ALTER and CHANGE MANAGER for DB2
User Guide: Volume 1

Supporting

Version 11.2.00 of BMC Object Administration for DB2
Version 11.2.00 of BMC Administrative Assistant for DB2
Version 11.2.00 of Database Administration for DB2
Version 11.2.00 of ALTER for DB2
Version 11.2.00 of CHANGE MANAGER for DB2

May 2015
Contacting BMC Software

Several methods are available for contacting BMC Software.

You can access the BMC Software website at http://www.bmc.com. From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

**United States and Canada**

**Address**
BMC SOFTWARE INC
2101 CITYWEST BLVD
HOUSTON TX 77042-2827 USA

**Telephone**
1 713 918 8800
or
1 800 841 2031

**Fax**
1 713 918 8000

**Outside United States and Canada**

**Telephone**
+01 713 918 8800

**Fax**
+01 713 918 8000

© Copyright 2007 - 2015 BMC Software, Inc.

BMC, BMC Software, and the BMC Software logo are the exclusive properties of BMC Software, Inc., are registered with the U.S. Patent and Trademark Office, and may be registered or pending registration in other countries. All other BMC trademarks, service marks, and logos may be registered or pending registration in the U.S. or in other countries. All other trademarks or registered trademarks are the property of their respective owners.

DB2, DB2 Universal Database, DRDA, IBM, OS/390, RACF, and z/OS are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

The information included in this documentation is the proprietary and confidential information of BMC Software, Inc., its affiliates, or licensors. Your use of this information is subject to the terms and conditions of the applicable End User License agreement for the product and to the proprietary and restricted rights notices included in the product documentation.

**Restricted rights legend**

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED—RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Section 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC SOFTWARE INC, 2101 CITYWEST BLVD, HOUSTON TX 77042-2827, USA. Any contract notices should be sent to this address.
Customer support

Support website
You can obtain technical support from BMC 24 hours a day, 7 days a week at http://www.bmc.com/support. From this website, you can:

■ Read overviews about support services and programs that BMC offers
■ Find the most current information about BMC products
■ Search a database for problems similar to yours and possible solutions
■ Order or download product documentation
■ Download products and maintenance
■ Report a problem or ask a question
■ Subscribe to receive proactive e-mail alerts
■ Find worldwide BMC support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by telephone or e-mail
In the United States and Canada, if you need technical support and do not have access to the web, call 1 800 537 1813 or send an e-mail message to customer_support@bmc.com. (In the subject line, enter SupID:yourSupportContractID, such as SupID:12345). Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC
Have the following information available so that Customer Support can begin working on your issue immediately:

■ Product information
  — Product name
  — Product version (release number)
  — License number and password (trial or permanent)
■ Operating system and environment information
  — Machine type
  — Operating system type, version, and service pack or other maintenance level such as PUT or PTF
  — System hardware configuration
  — Serial numbers
  — Related software (database, application, and communication) including type, version, and service pack or maintenance level
■ Sequence of events leading to the problem
■ Commands and options that you used
■ Messages received (and the time and date that you received them)
  — Product error messages
  — Messages from the operating system
  — Messages from related software
License key and password information

If you have questions about your license key or password, contact Customer Support through one of the following methods:

- Send an e-mail message to customer_support@bmc.com. (In the Subject line, enter SupID:yourSupportContractID, such as SupID:12345.)
- In the United States and Canada, call 1 800 537 1813. Outside the United States and Canada, contact your local support center for assistance.
## Contents

### About this book
- Related publications ................................................................. 9
- Conventions ................................................................................. 10
- Syntax statements ........................................................................ 10

### Chapter 1  Overview of ALTER and CHANGE MANAGER
- Change management ........................................................................... 13
  - Identification of your environment ........................................... 13
  - Location of your environment ................................................... 14
  - Propagation of your changes ....................................................... 14
- Overview of change management with ALTER and CHANGE MANAGER ........................................ 15
  - Overview of ALTER .................................................................. 15
  - Overview of CHANGE MANAGER .............................................. 19
- ALTER architecture ........................................................................ 23
  - ALTER components .................................................................... 25
  - ALTER objects ........................................................................... 28
- CHANGE MANAGER architecture .................................................. 30
  - CHANGE MANAGER components ........................................... 33
  - CHANGE MANAGER objects .................................................... 36
- Solution integration ......................................................................... 44
  - Administrative Assistant for DB2 ............................................. 44
  - Database Administration for DB2 ............................................. 44
  - BMC Object Administration ..................................................... 45

### Chapter 2  Overview of the ALTER and CHANGE MANAGER interface
- Before you use the product ............................................................. 47
- ALTER and CHANGE MANAGER interface ................................ 47
  - Main Menu .................................................................................. 48
  - Displaying the current product environment .................................. 49
  - Using Fast Path Navigation ......................................................... 50
  - ISPF commands ......................................................................... 51
  - BMC message information ....................................................... 53
- Achieving your goals with ALTER and CHANGE MANAGER ............ 53
  - Creating, altering, or dropping DB2 objects .................................. 54
  - Migrating DB2 objects ............................................................... 86
### Chapter 3 Setting up ALTER and CHANGE MANAGER

#### Using options to control your environment
- Using the installation options .................................................. 138
- Using the user options ............................................................... 139
- Using the product options ......................................................... 140
- Product option overrides .......................................................... 141
- Additional installation options module ........................................ 142
- How ALTER or CHANGE MANAGER use options ....................... 143

#### Data set allocation ............................................................... 144

#### Object names ................................................................. 144
- Wildcard patterns ..................................................................... 145
- Delimited identifiers ................................................................. 147

#### Nonprintable or nonviewable characters ................................ 148

#### Access to components ....................................................... 149
- ALTER and CHANGE MANAGER plans .......................................... 149
- CHANGE MANAGER plans ....................................................... 150
- Execution component plans ...................................................... 151

#### Authorization switching ...................................................... 152
- Execution security exit function codes ......................................... 153
- Execution security exit testing .................................................... 155

#### Access to the worklist parallelism feature ............................... 156
- Zap to restrict access to worklist parallelism ................................. 156
- General resource profile for worklist parallelism .......................... 157

#### Setting the MEMLIMIT system parameter .............................. 157

#### Modification of ISPF skeletons and variables ........................ 158

#### JCL Generation data sets sizing function ............................... 159

#### Product table maintenance ................................................... 160
- Table size maintenance ............................................................. 164
- Table security .......................................................................... 165

#### Setting options .................................................................... 168
- Setting the attributes for the panel display ................................. 168
- Setting the options for data set allocation ................................. 170
- Setting the processing options for Analysis ............................... 171
- Setting the processing options for Import ................................. 173
- Setting the processing options for Execution ............................. 174
- Setting the processing options for Compare .............................. 176
- Setting the processing options for Baseline ............................... 177
- Setting the options for the baseline report ............................... 178
- Setting the user options for CM/PILOT ....................................... 179
- Setting the JCL options for job cards ....................................... 182
About this book

This book contains detailed product information and is intended for application developers and database administrators (DBAs).

Like most BMC documentation, this book is available in printed and online formats. To request printed books or to view online books and notices (such as release notes and technical bulletins), see the support website at http://www.bmc.com/support.

**Note**

Online books are formatted as PDF or HTML files. To view, print, or copy PDF books, use the free Adobe Reader from Adobe Systems. If your product installation does not install the reader, you can obtain the reader at http://www.adobe.com.

The software also offers online Help. To access Help, press F1 within any product or click the Help button in graphical user interfaces (GUIs).

Related publications

From the BMC Support Central website, you can use the following methods to access related publications that support your product or solution:


- View Quick Course videos (short overviews of selected product concepts, tasks, or features), which are available from the following locations:
  
  — Documentation Center
  
  — Support Central (at http://www.bmc.com/support/mainframe-demonstrations)
  
  — BMC Mainframe YouTube channel (https://www.youtube.com/user/BMCSoftwareMainframe)

You can order hardcopy documentation from your BMC sales representative or from the support site. You can also subscribe to proactive alerts to receive e-mail messages when notices are issued.

**Tip**

You can access the BMC Support Central site at http://www.bmc.com/support.

---

## Conventions

This document uses the following special conventions:

- All syntax, operating system terms, and literal examples are presented in this typeface.
- Variable text in path names, system messages, or syntax is displayed in italic text: `testsys/instance/fileName`
- Menu sequences use a symbol to convey the sequence. For example, Actions => Create Test instructs you to choose the Create Test command from the Actions menu.

## Syntax statements

This topic explains conventions for showing syntax statements.

A sample statement follows:

```
COMMAND KEYWORD1 [KEYWORD2 | KEYWORD3] KEYWORD4={YES | NO} fileName...
```

The following table explains conventions for syntax statements and provides examples:
<table>
<thead>
<tr>
<th><strong>Convention</strong></th>
<th><strong>Example</strong></th>
</tr>
</thead>
</table>
| Items in italic type represent variables that you must replace with a name or value. If a variable is represented by two or more words, initial capitals distinguish the second and subsequent words. | alias  
databaseDirectory  
serverHostName |
| Brackets indicate optional items. Do not type the brackets when you enter the option. A comma means that you can choose one or more of the listed options. You must use a comma to separate the options if you choose more than one option. | [tableName, columnName, field]  
[-full, -incremental, -level] |
| Braces indicate that at least one of the enclosed items is required. Do not type the braces when you enter the item. | {DBDName | tableName}  
UNLOAD device={disk | tape, fileName | deviceName}  
{-a | -c} |
| A vertical bar means that you can choose only one of the listed items. In the example, you would choose either commit or cancel. | {commit | cancel} |
| An ellipsis indicates that you can repeat the previous item or items as many times as necessary. | columnName... |
Overview of ALTER and CHANGE MANAGER

As DB2 applications become more structurally complex and more mission critical, the need to add and modify data structures, as well as the need for change management, increases significantly.

In addition, the growing complexity of the DB2 environment has made the change process itself more difficult and costly. Without an effective change management tool, database administrators (DBAs) find that data structure changes and change management is complex, tedious, resource intensive, and error prone.

For more information, view the Quick Course "CHANGE MANAGER for DB2 - Getting Started."

Change management

Every organization, every environment, and every application is unique. Before you can use change management tools to help you manage changes to your data structures and your data, you might need to consider the following questions:

- What does my environment look like?
- Where are my environments located?
- How are my changes propagated?
- When are my changes propagated?

Identification of your environment

Most organizations use at least three environments for each of their applications: development, test, and production.
What does your environment look like? Perhaps you maintain your application code and perform unit tests in a development system. Your test system might be used to perform system and stress tests, as well as simulate production. Your production system might be in single or multiple locations.

**Location of your environment**

Sometimes multiple environments are located on the same DB2 subsystem.

The same DB2 subsystem can be used for multiple environments if you use different database names and you have different owners. Where are your environments located? Are they located on the same subsystem? Do they share DASD? Are they located in different cities or in different parts of the world?

**Propagation of your changes**

In some organizations, multiple teams are responsible for developing or maintaining an application in a development environment.

If the changes to the data structures are needed by each team, you might need to synchronize the data structures in the development environment before you migrate the entire structure to the test environment. How will you propagate these changes? Do you manage your change requests based on release cycles or by date?

One way to manage your development, test, and production environments is as follows:

- If you need to create an environment, you can migrate your data structures and data.
- If the environment already exists, you can compare one environment to another to determine what changes need to be made to synchronize the data structures in both of the environments.
- You can create, alter, or drop data structures in the environment, or simply maintain the environment.
For example, you could create your development environment based on your production environment or synchronize your development environment with your production environment. To create your development environment, you could migrate your production environment. To synchronize your development environment with your production environment, you could compare your development environment to the production environment. After your development environment is established, you could modify it or maintain it.

**Overview of change management with ALTER and CHANGE MANAGER**

BMC provides two products to help you manage data structures and data for DB2 subsystems: the ALTER and CHANGE MANAGER products.

**Overview of ALTER**

ALTER provides a powerful solution to the problems of managing a DB2 environment.
By automating and simplifying the change process, ALTER enables you to deal effectively with the demands of a constantly changing DB2 environment. ALTER uses an Interactive System Productivity Facility (ISPF) interface.

ALTER is designed for the single DB2 system that is beginning to encounter complex challenges. As the need for more change management functions develops, as with multiple DB2 subsystems, you can upgrade from ALTER to CHANGE MANAGER.

ALTER enables you to automate the following common tasks:

- Create, modify, and drop data structures within a DB2 subsystem
- Create new data structures by using existing data structures as templates
- Determine the dependencies on changed objects and preserve those dependencies and their associated data
- Migrate data structures and data within a subsystem or from one DB2 subsystem to another subsystem
- Analyze the impact of changes by validating the changes against the DB2 catalog. If conflicts are detected, ALTER notifies you through error messages and warnings.
- Implement a least-cost strategy for performing changes
  - You can group your changes to minimize the work that is required to execute them.
- Import data definition language (DDL) files and create the objects
- Perform space estimation for table spaces and indexes
- Provide passive management of authorizations, plans, and packages
- Provide automatic data conversion for changes to column attributes
- Handle requirements for data set allocations
- Specify global structure changes when migrating objects

ALTER provides full management support for the following DB2 data structures:

- Storage groups (stogroups)
- Databases
- Table spaces
When you use ALTER to specify changes for any of these data structures, the product automatically propagates the changes to any dependent objects. For example, if you change the name of a table, ALTER creates a corresponding change in the indexes, synonyms, and other dependent objects that reference the table under its former name.

ALTER uses BMC or IBM utilities in worklists when required. By using installed BMC utilities instead of IBM utilities, you can significantly enhance the performance of executing ALTER worklists. BMC utilities run faster, provide additional features, and might reduce the number of steps in a worklist.

You can use the following BMC utilities with ALTER:

- BMCSTATS utility of DASD MANAGER PLUS for DB2
- COPY PLUS for DB2
- CHECK PLUS for DB2
- LOADPLUS for DB2
- RECOVER PLUS for DB2
- REORG PLUS for DB2
- UNLOAD PLUS for DB2

The utilities must support the same version of DB2 as ALTER or CHANGE MANAGER. For more information, see the utility product’s reference manual.
Data structure changes

Within a DB2 subsystem, an application’s data structures need continual modification and extension because of changes in application requirements, performance tuning, and expanding uses of data.

ALTER enables you to make changes to data structures in an easy, reliable, and automated way. You can efficiently perform changes while preserving data and, if necessary, convert the data to match the changes. ALTER also enables you to rebuild dependent objects and then propagate changes into those dependent objects. Without ALTER, these tasks can be extremely difficult to perform.

WARNING

BMC does not recommend using ALTER to modify the DB2 catalog, directory, and workfile objects.

Data structure migration

If you have multiple DB2 subsystems, you might need to copy data structures from one subsystem to another.

Most DB2 sites have separate subsystems for development, testing, and production. Many sites support multiple production systems, some of which are local and some remote.

ALTER offers the following functionality to simplify data structure migration:

- Generates SQL CREATE statements from the sending subsystem for use on the receiving subsystem
- Applies CREATE statements on the receiving subsystem
- Moves data from the sending subsystem to the receiving subsystem

When you migrate an application’s data structures, ALTER assumes that these data structures do not yet exist on the receiving subsystem. ALTER cannot create new data structures if data structures with the same names already exist on the receiving subsystems.

Data structure importation

You use ALTER to process DDL files that are received from an external subsystem and apply the changes to your subsystem.
The Import component converts these data structure definitions into change requests that are grouped together in a work ID. ALTER stores these change requests in the Change Definition (CD) tables.

**Space estimation**

ALTER provides space estimation capability that enables you to determine the amount of space that a table space or index will require based on the object definitions and their estimated usages.

You can specify estimates of the number of rows in the table, and the average length of the rows. ALTER will use this information to project the number of tracks, cylinders, or blocks that are required to allocate the data set for that object.

**Note**

You cannot resize the DB2 catalog or directory tables.

**Overview of CHANGE MANAGER**

CHANGE MANAGER enables the DBA to effectively deal with the demands of a constantly changing environment involving multiple DB2 subsystems. Like ALTER, CHANGE MANAGER uses an ISPF interface.

CHANGE MANAGER includes all of the features in ALTER and provides the following additional capabilities:

- Migrates data structure changes across multiple databases and subsystems
- Determines changes to data structures and migrates those changes to one or more copies of the data structures
- Captures and records structure definitions and data within a DB2 subsystem at a point in time (to establish a full-recovery baseline)
- Recovers structures and data to a point in time defined by a full-recovery baseline within a DB2 subsystem
- Compares two versions of structure definitions to:
  - Determine the changes necessary to upgrade one version to another
  - Selectively apply changes to copies of the data structures while preserving the uniqueness of each copy
- Uses data modeling tool outputs to determine the changes to existing DB2 application structures
- Reduces the volume of information that is needed to communicate changes by using a BMC language called Change Definition Language (CDL) to transmit the change information
- Feeds changes that are made on a remote system back to the development system
- Defines and stores reusable rules for making changes
- Uses SQL-like statements to update, delete, and migrate data structures

CHANGE MANAGER provides full management support for the same DB2 data structures as ALTER does.

**Data structure change migration**

*Change migration*, which means to migrate only data structure changes (instead of entire structures) to another copy of the structure on the same DB2 subsystem or on a different DB2 subsystem, enables you to update structures that have already been migrated to another copy.

However, you must also be able to retain any structure modifications that were made locally after the structures were migrated. DB2 users often make local modifications—or variations from the control definition of the application—to meet the following needs:

- **Performance and usage tuning**
  Local tuning is a complex process that is based on the specific system that is involved and the performance and use of the application in that environment. Because separate sites often have different amounts of data and different transaction loads, as well as different CPU and DB2 environments, local tuning of application structures is common.

- **Security**
  Authorization requirements are associated with object ownership and, in some cases, with object names. As a result, the authorizations for one DB2 subsystem can vary significantly from the authorizations of another subsystem running the same application. In this situation, each subsystem requires local modifications to the application.

- **Additional uses of data**
  Often, production systems in a DB2 environment need to use data that is beyond the fixed scope of an application. For example, the data in a payroll application might be needed for reports that the finance department develops. DB2 enables
users to locally generate additional data structures (such as indexes, synonyms, views, aliases, and authorizations) to meet these needs.

Because local modification of applications is so common in DB2, users need an application management strategy that enables them to manage basic elements of the application globally, without compromising the elements that vary locally.

CHANGE MANAGER satisfies the requirement of migrating global changes while retaining local modifications by:

- Determining the changes made to the control version of the application
- Applying only those changes to other versions while preserving all local modifications

CHANGE MANAGER can also determine the accuracy of the proposed changes before applying them. Because changes are allowed at the lowest level of an object’s definition, CHANGE MANAGER preserves and converts data as necessary to accommodate the requested changes.

Additionally, CHANGE MANAGER propagates changes to all dependent structures. For example, when you change a column name, CHANGE MANAGER propagates the change to any index and foreign key definitions that use the column, even though the index and foreign key definitions might be unknown to the sending DB2 subsystem.

**WARNING**

BMC does not recommend using CHANGE MANAGER or the CM/PILOT component of CHANGE MANAGER to modify the DB2 catalog, directory, and workfile objects.

### Data structure recovery

Changes to data structures might not produce the effect that you intended.

For example, a change might significantly increase an application’s response time. In this case, you might need to restore the previous data structures. While DB2 has features for logging, backup, and recovery of data, it has no similar features for data structures. If you make an unwanted structure change, or if your changes fail or are unusable, you must fall back to the previous definition. To do so, you must have saved the structure definitions and data of the previous version.

CHANGE MANAGER enables you to capture a set of data structures from the DB2 catalog or DDL file and store the set in a baseline. A baseline can contain only data structures or data structures with their associated data.
Establishing baselines and recovering to them is called data structure recovery. You can reload the data that is stored in a full-recovery baseline after the structures have been recovered. You can also reload the data that exists in those structures at the time of their recovery. In addition, CHANGE MANAGER can convert current data to match the restored data structures.

Change recording and controlling

Managing change also requires that you know what changes have been made to an application’s data structures during different periods of time.

CHANGE MANAGER provides a comparison feature that can generate a file that shows the differences between two sets of data structures. Because the file is stored in a sequential data set or partitioned data set (PDS) member, you can view, edit, print, or store it.

Change feedback

Data structure changes might not flow in order from the development system, through testing, and into production.

Because changes to the basic application definition can occur at any point in the cycle, you must be able to transmit changes back to the development system or to the control node. Transmitting changes from a remote system to the development system (or to the control node) is called change feedback.

To feed back changes, CHANGE MANAGER enables you to:

- Identify the changes made
- Apply the changes at the development or control system

You do not need to feed back all of the changes to the development system. You should feed back only those changes that are part of the globally managed basic definition of the application.

Data modeling tools

In many installations, the data structure design of the application is not controlled in a DB2 subsystem but in a data modeling tool (or Computer-Aided Software Engineering, or CASE, tool), data dictionary system, or other structure design repository.

In this guide, the term data modeling tool refers generically to any external design tool.
Most data modeling tools generate an application’s DDL. Changing an application’s data structures generates new DDL, which you can use to create the data structures in the DB2 catalog. Although data modeling tools can improve the productivity of DB2 developers, the use of these tools creates the following problems:

- If you make an emergency change in a production system, including that change in the data modeling design is difficult and usually requires a manual entry. You can use host-based DB2 tools to make emergency changes (or to define local structure elements not supported by a data modeling tool), but the changed structures will no longer be synchronized with the data modeling design.

- Often, communication between a data modeling tool and DB2 is one way. You can design structures on the data modeling tool and transmit the structures to DB2, but updating the data modeling design after making changes at the DB2 node is difficult.

- Most data modeling tools cannot selectively modify data structures. Because these tools can transmit only a complete application definition, they might be useful only for the initial design of an application. Maintenance and upgrades are then performed at the DB2 level.

- Due to their ease of use and built-in design intelligence, data modeling tools are popular for managing global application data structure definitions. However, the external DB2 specifications that a data modeling tool produces might be incomplete or might fail to reflect local elements of the design.

You can address these problems by using CHANGE MANAGER to migrate changes between the data modeling definition of an application and the DB2 definition. CHANGE MANAGER enables you to:

- Establish control points for the application’s data structures and determine the changes that are made between those control points

- Implement design changes in DB2 without losing either the application’s data or its local data structure modifications

- Transmit changes from DB2 back to the data modeling tool after making global changes to the DB2 version of the application’s data structures

**ALTER architecture**

ALTER provides management of data structures and data for a DB2 subsystem.

The product consists of several components that allow you to create, modify, and drop DB2 data structures while preserving data and dependent objects.
Figure 1 on page 24 illustrates the architecture of the ALTER product. The components and the objects of ALTER are discussed in the sections that follow.

**Figure 1: ALTER architecture**
Note
Although not depicted in Figure 1 on page 24, the Front End component interacts with other ALTER objects and components. For more information, see “ALTER Front End” on page 25.

ALTER components

Most of the components that ALTER uses to perform its tasks can run in the foreground or as batch jobs.

ALTER consists of the following components:

- Front End
- Specification
- Import (DDL only)
- Analysis
- JCL Generation
- Execution

ALTER Front End

The Front End component is a dialog panel that you use to access and control ALTER.

From the Front End component, you can create, analyze, and execute work IDs, and build JCL in foreground or batch mode.

ALTER Specification

The Specification component enables you to create or edit data structure change or migration requests.

The component stores the change or migration requests in a work ID. With Specification, you can perform the following tasks:

- Create data structures
- Change data structures
Delete data structures

Migrate data structures

Estimate the space required for a table space or index

The Specification component lists objects from the DB2 catalog (or from a copy of that catalog if your site uses catalog indirection). You can change these objects, delete them, or create new objects either from the list of objects or from the Command line of some Specification panels. Changes are not made to the data structures until you execute the change and migration requests.

**ALTER Import**

You can import data structure definitions that are stored in a DDL file.

The Import component converts these data structures into change requests in a work ID. The Import component provides a batch function similar to that provided by the Specification component with limited validation checking.

**ALTER Analysis**

After you create change or migration requests in a work ID, the Analysis component checks the requests for validity with the DB2 catalog, develops an optimal implementation strategy, and generates a worklist.

The worklist contains the DB2 utility commands, AMS commands, security commands, and SQL statements that are necessary for implementing the requests. In addition, the Analysis component propagates changes to dependent structures. The Analysis component can be run in the foreground or in batch using the JCL Generation component.

The Analysis component performs the following validity and compliance checks:

- Verifies that the data structures of changing objects have not changed in the DB2 catalog since the present changes were requested

- Checks the change requests against other requested changes and warns of any conflicts

- Checks the change requests for compliance with DB2 rules and reports conflicts

- Develops the most efficient strategy to implement the requested changes or migrations

- Locates all dependent structures
- Binds or rebinds plans and packages
- Migrates plan and package authorizations
- Checks for objects that are indirectly affected by requested changes

**Note**
The Analysis component warns you if a drop or alter change request will result in the loss of objects, and also warns you of views and plans that might require modifications.

If no errors are found, the Analysis component generates a worklist, stores the worklist or associated data set name with the work ID, and uses it as the default worklist when you run the Execution component. The Analysis component also creates diagnostic output in ALUPRINT and provides authorization switching.

**ALTER JCL Generation**

The JCL Generation component works with the Front End component to construct a job control language (JCL) file for running the Analysis and Import components in batch.

Using symbolic variables, the Front End component resolves all data set names that appear on the interface panels of the components. These names include the names of input files, JCL files, and the diagnostic output files.

**Execution functions**

The Execution component performs the commands that are contained in the worklist that the product generates for change or migrate requests.

Although the tasks that are required to implement a worklist vary depending on the complexity of the actions involved, the Execution component performs the following basic functions:

- Performs worklist control-command processing for DASD MANAGER PLUS
- Unloads DB2 data
- Runs IBM or BMC utilities
- Executes DB2 commands and Access Method Services (AMS) statements
- Establishes synchronization (sync) points to accommodate restart, if needed
- Creates diagnostic output in AEXPRINT
- Performs audit and recovery logging, if requested
- Performs authorization switching to ensure that DB2 objects are created with the correct owner and creator

The Execution component is the only component that makes physical changes to DB2 data structures.

**ALTER objects**

ALTER uses the following objects to perform its tasks.

This section describes each of the following objects in detail:

- Work IDs
- Worklists
- Unload data sets
- DDL files
- Internal tables

**Work IDs**

Work IDs enable you to identify, control, and track a unit of work.

Work IDs contain change or migration requests that are stored in the CD tables. Creating or selecting a work ID is usually the first step in performing any type of change or migrate process.

ALTER uses the following types of work IDs:

- Alter-type
  
  You use an alter-type work ID to change or modify data structures, drop objects, or create new data structures.

- Migrate-type
  
  You use migrate-type work IDs to specify the data structure definitions that are to be migrated from one DB2 subsystem to another, or to duplicate structures within the same subsystem. Migrate-type work IDs do not modify existing data structures.

  Migrate-type work IDs can contain change rules to modify database object attributes and migrate options to specify the default dependencies to include in a
migration. The change rules compare the object type and attribute to those in the migrated data structures and apply the rule when a match is found.

The migrate process provides a way to create new data structures from data structures that already exist on a subsystem. You can create the new data structures on the same subsystem if the objects are renamed. You can rename the objects through the Specification component or with change rules in the work ID.

- Receive-type
  You use receive-type work IDs to receive a worklist that is generated by a migrate-type work ID that contains the migrated data structures and data from a different subsystem.

You can request any number of changes or migrations with a single work ID. You can enter these requests all in one session or across many sessions. You can also delete or modify requests if you make a mistake or change your mind. Any number of users can request changes or migrations with the same work ID.

For more information, view the Quick Course "CHANGE MANAGER for DB2 - Creating Work IDs."

ALTER worklists

The worklist data set contains all of the commands and SQL statements that are needed to execute the changes or migrations specified in a work ID.

The Analysis component generates the worklist commands in a specific order. You can edit the worklist, but changing the order of the commands might yield incorrect results.

The worklist can contain any of the following items:

- SQL statements that perform the change or migration requests, including the dropping and re-creating of all affected objects and the restoring of dependent objects that are lost as a result of a change or that propagate the changes into other objects

- DB2 and utility commands that implement requested changes and migrations, including:
  - Unloading data
  - Reorganizing table spaces and indexes
  - Reloading data
  - Collecting statistics
— Making image copies
— Rebuilding indexes
— Checking the data
— Rebinding application plans and packages (if an alter-type work ID)
— Binding plan and package authorizations (if a migrate-type work ID)
— Performing other authorizations

■ AMS commands for data set allocation and deletion that support VCAT-defined partitions

**Unload data sets**

ALTER uses unload data sets to store data while dropping and rebuilding DB2 objects.

**ALTER DDL files**

ALTER does not generate DDL files. ALTER can use DDL files to import data structures.

**ALTER internal tables**

ALTER uses the following types of internal tables to perform its tasks:

■ CD tables, which store change and migration specifications for each work ID
■ CM tables, which store work ID, change rule, and sync data

**CHANGE MANAGER architecture**

CHANGE MANAGER consists of all of the ALTER components, as well as additional components that allow you to perform tasks across multiple subsystems.
Figure 2 on page 31 illustrates the architecture of the CHANGE MANAGER product. The components and the objects of CHANGE MANAGER are discussed in the sections that follow.

**Figure 2: CHANGE MANAGER components and objects**
The shaded icons represent the components and objects that are unique to CHANGE MANAGER. The CM/PILOT component of CHANGE MANAGER and the interaction of the Front End component with other CHANGE MANAGER objects and components is not depicted. For an illustration of the CM/PILOT component, see Figure 3 on page 32. For more information about the interaction of the components, see “ALTER components” on page 25.
Note
The shaded icons in Figure 3 on page 32 represent the components and objects that are unique to the CM/PILOT component of CHANGE MANAGER. CM/PILOT also uses some of the components and objects of CHANGE MANAGER. For an illustration of the CHANGE MANAGER architecture, see Figure 2 on page 31.

CHANGE MANAGER components

Most components that CHANGE MANAGER uses to perform its tasks can run in the foreground or as batch jobs.

CHANGE MANAGER uses the following components:

- Front End
- Specification
- Baseline
- Compare
- Import
- Analysis
- JCL Generation
- Execution
- CM/PILOT

The Specification and Analysis components provide the same functionality as they do in ALTER. For a description of these components, see “ALTER components” on page 25.

CHANGE MANAGER Front End

In CHANGE MANAGER, the Front End component can also maintain migrate and baseline profiles, and interact with the Baseline, Compare, and Import components.

Baseline

The Baseline component captures a set of DB2 structure definitions from either the DB2 catalog, a DDL file, or a migrate worklist at a specific point in time.
The captured set of data structure definitions is called a baseline. If the structures are defined in the DB2 catalog, a baseline can also capture the data and authorizations that are associated with those structures. A baseline of both data structures and data is called a full-recovery baseline.

Baselines act as control points during data structure management. Baselines establish a static set of data structures for an application version. If you make a change with unwanted results, you can restore the data structures. If you establish a full-recovery baseline, you can also restore the data. You can also restore the data structure back to a prior baseline and convert the current data to those structures.

You can also use baselines for version comparison. For example, when you first install an application, create a baseline. At a later time, perform a comparison between the baseline and the DB2 catalog. This comparison creates a CDL file that shows any modifications that have been made to the application’s data structures since the initial installation.

**Note**
A baseline is also a CHANGE MANAGER object.

**Compare**

The Compare component determines the differences between two sets of data structures and then generates a CDL file.

The generated CDL file contains all of the changes that the comparison found between the two sets of data structures. You can compare data structures that are stored in a DDL file, work ID, baseline, worklist, or DB2 catalog. The inputs to the Compare component are called the primary and secondary input sources. The CDL file contains the changes that, if applied to the primary input source, would produce the data structures of the secondary input source.

You can use the generated CDL file in any of the following ways:

- Process the file as a set of change requests for the current subsystem
- Save the file as a record of the changes made
- Import the file to a different subsystem in order to update a separate version of the data structures

CDL has advantages over DDL in automating the process of updating data structures. CDL allows more types of modifications to data structures and, unlike DDL, can retain local modifications to those structures.
**CHANGE MANAGER Import**

You can import data structures that are stored in a CDL, DDL, or DML file.

The Import component converts the data structures in the files into change requests in a work ID. CHANGE MANAGER stores these change requests in the CD tables. The Import component provides a batch function similar to that provided by the Specification component with limited validation checking.

**CHANGE MANAGER JCL Generation**

In CHANGE MANAGER, the JCL Generation component also generates JCL for the Compare, Baseline, Baseline Report, and CM/PILOT components.

**CHANGE MANAGER Execution**

When CHANGE MANAGER is used in the Database Administration for DB2 solution, the Execution component can enable a worklist to execute in parallel.

That is, the component executes portions of the worklist concurrently. For information about the Database Administration for DB2, see “Database Administration for DB2” on page 44.

**CM PILOT**

The CM/PILOT component automates DB2 change management tasks. With CM/PILOT, you do not need to decide which CHANGE MANAGER processes are required for a task or the sequence in which you need to complete them—CM/PILOT provides scripts to guide you through the process.

The CM/PILOT component enables you to choose from several predefined scripts or to create your own scripts. You can copy scripts, including those that CM/PILOT provides, and modify them to meet your needs. You can also edit, browse, delete, and perform other maintenance tasks on scripts.

CM/PILOT enables you to quickly and easily use the vast change management power of CHANGE MANAGER. By following the dialog panels that are provided in the CM/PILOT scripts or in the scripts that you create, you can specify, analyze, and execute an Analysis worklist to perform the following tasks:

- Change data structures
- Migrate data structures
- Migrate only data
- Receive data structure changes
- Receive DDL to create data structures
- Create full-recovery baselines
- Recover data structures with current or old data
- Select a user-defined script
- Replicate work IDs

When you want to perform a change management task, CM/PILOT determines the components that CHANGE MANAGER will use and ensures that they are run in the correct sequence. CM/PILOT enables you to create tasks that can be done later by someone else or through job scheduling. By reusing these tasks, you can ensure that the change management task is done the same way every time.

The CM/PILOT component enables you to easily prepare for various CHANGE MANAGER tasks that can be processed on demand. CM/PILOT dialog panels are task oriented and easy to use. The panels guide you through the CHANGE MANAGER components that are necessary to accomplish the specific tasks and prompt you for the information that is needed to build worklists that can be processed at any time.

CM/PILOT simplifies the decision-making process and enables you to prepare change management tasks that are performed repeatedly. It also lets you create change management tasks that can be performed by less-experienced DBAs. And with its unique script-locking feature, CM/PILOT lets you create script steps that cannot be modified and can only be run as you have specified.

You can also use two CM/PILOT scripts to help you create SQL-like Data Manipulation Language (DML) statements to update, delete, and migrate data structures. You can also import DML statements into an alter-type work ID. After the statements are imported, CHANGE MANAGER creates entries in the CD tables as if they had been requested in the Specification component.

**CHANGE MANAGER objects**

CHANGE MANAGER uses the following objects to perform its tasks.

This section describes each of the objects in detail.

- Work IDs
- Worklists
The work ID and unload data set objects are the same objects as in ALTER. For a description of these objects, see “ALTER objects” on page 28.

**CHANGE MANAGER worklists**

When CHANGE MANAGER is used in the Database Administration solution, the worklist can also contain keywords that control the sequencing of parallel processes.

For information about the Database Administration solution, see “Database Administration for DB2” on page 44.

**Baselines**

A baseline contains a set of data structures and their associated authorizations, taken at a specific point in time, from the DB2 catalog, a DDL file, or a worklist.

CHANGE MANAGER uses structure-only and full-recovery baselines:

- A structure-only baseline contains only data structures.
- A full-recovery baseline contains data structures and data.

The types of input to a baseline are as follows:
- **DB2 catalog**
  A catalog baseline takes its input from a subset of the DB2 catalog. See “Baseline profiles” on page 38.

- **DDL file**
  A DDL baseline takes its input from a file containing SQL DDL statements.

- **Worklist file**
  A worklist baseline takes its input from a worklist that CHANGE MANAGER or another BMC product produces. A worklist baseline is identical to a DDL baseline except for the input source file type. See “Baseline profiles” on page 38.

## Profiles

Profiles are a collection of scope rules, change rules, and locations that enable you to define and control the Analysis, Compare, and Baseline processes.

You use profiles to:

- Select a set of objects
- Customize changes to objects that are migrated to other subsystems
- Create baselines of application data structures

Profiles enable you to automate changes instead of selecting objects individually for migration or making individual changes to object structure definitions. You can create, copy, edit, and reuse profiles.

Two types of profiles are used in CHANGE MANAGER: baseline profiles and migrate profiles.

### Baseline profiles

You use baseline profiles to create, name, and maintain baselines of application data structures.

You can automatically manage the baselines that you want to keep and those that you want to delete by specifying delete and retain values in the baseline profile. The two types of baseline profiles are catalog and DDL (or worklist) profiles.

- **Catalog baseline profiles**
  The profile for a catalog baseline must contain scope rules that select the DB2 objects to include in the baseline or comparison. The baseline profile can explicitly
specify the scope rules or can reference the scope rules in an outbound migrate profile.

Scope rules are used only when a process is being performed on the DB2 catalog. The scope rules serve to create a subset of the DB2 catalog.

■ DDL or worklist baseline profiles

When defining a DDL baseline profile, you define only header information. The scope is implicitly defined by the set of objects in the DDL or worklist.

**Migrate profiles**

You use migrate profiles to select a set of objects and to customize changes to objects that are migrated to different locations.

The two kinds of migrate profiles are inbound and outbound.

■ Inbound Migrate Profiles

The Import component uses inbound migrate profiles to make input from another system compatible with the receiving subsystem’s version of the application. Inbound migrate profiles contain only change rules that modify imported data structure definitions.

The change rules are applied as the CDL file or DDL file is imported. For example, if a change rule specifies changing the database name DEMO* to ACM*, the DDL statement CREATE DATABASE DEMOHRS is read and the CD table entries are generated with a CREATE DATABASE ACMHRS. The profile’s change rules are applied to the imported data structure definitions before CD table entries are created.

You cannot specify scope rules or locations with an inbound migrate profile. The file being imported identifies the scope.

■ Outbound Migrate Profiles

You use outbound migrate profiles to generate worklists or CDL files. The outbound migrate profile can contain the following items:

— A scope for specifying the DB2 objects to include in the operation

— Locations for defining multiple clones of an application

— Change rules for defining attribute changes to be automatically applied to objects in each clone

You can group locations if they reside in the same DB2 subsystem.

When an outbound migrate profile is used as input to Analysis, a worklist is generated for each group specified. One CDL output file is generated for each...
group when an outbound migrate profile that specifies locations is used as an input to the comparison.

The Compare process can also use the scope rules of an outbound migrate profile to select a subset of objects from the DB2 catalog to use in a comparison with a DDL file or worklist.

For more information, view the Quick Course "CHANGE MANAGER for DB2 - Creating Migrate Profiles."

**Scope rules**

When you define scope rules for a profile, you are specifying which objects to fetch from the catalog.

You can reference scope rules when you perform a comparison, create a baseline, or analyze a work ID.

In a catalog baseline profile, scope rules specify the DB2 objects to include in the baseline. The scope rules of an outbound migrate profile can specify the objects to migrate or the DB2 objects to include in a comparison. Scope rules can include DB2 objects in the final set of objects to be processed or can exclude them from the final set.

Scope rules can select a specific object or select groups of objects (through the use of wildcard characters) as the set or sets of objects on which to operate. Scope rules can also determine which dependent objects to include and to exclude. For example, a scope rule could include database ABCDEF and its dependent table spaces, tables, indexes, foreign keys, and views, but exclude its dependent synonyms, aliases, or data.

You can define scope rules for both migrate profiles and catalog baseline profiles. You can follow the same procedure to create a set of scope rules for each type of profile. You can also use the same set of scope rules for different profiles. For example, when creating a catalog baseline profile, you can reference the scope rules of a migrate profile. Likewise, a migrate profile can reference the scope rules of a catalog baseline profile. This feature enables you to use the same list to select the same objects for either migration or a baseline.

The Compare component uses the scope rules of a catalog baseline profile when it compares a catalog baseline to the DB2 catalog. The Compare component finds the baseline profile by using the established baseline name, and then uses the scope rules of the catalog baseline profile to select the objects to compare. Compare can also use the scope rules of a specified profile when it compares a DB2 catalog to a DDL file or to a worklist.
Change rules

Change rules modify the attributes of existing DB2 object structure definitions.

The change rules compare the object type and attributes to those in the migrated data structures. The change rules are applied when the attributes and the names match. For ALTER, you can define change rules within a migrate-type work ID. For CHANGE MANAGER, you can define change rules within a migrate-type work ID or a migrate profile.

When you define change rules within a migrate profile, the rules are processed for each object that is defined in the scope of the profile. You can streamline and automate changes to sets of object structures by defining change rules for migrate profiles. Because migrate profiles are used in different types of migration processes, special consideration of how change rules work can significantly contribute to the efficient performance of these processes.

Change rules allow you to easily specify and repeat common changes. For example, in many DB2 installations, the owner of a table changes when the table is moved from a test system into production. A migrate profile can perform the changes that are necessary to migrate the structures from the test system into production. This profile can include a change rule that changes all of the tables with owners that match TEST to PROD.

Change rules can perform the following tasks:

- Change the names of objects
- Change the attributes of a specified object or group of objects
- Include specific volumes in a storage group, or exclude volumes from a storage group
- Exclude one or more columns from a table
- Specify that table spaces and indexes that have a STOGROUP attribute be defined to use VCAT on the receiving database
- (Compare only) Suppress changes to altered objects

Locations for outbound migrate profiles

Locations, which are defined only in outbound migrate profiles, enable you to create or maintain multiple clones of an application schema.

When an outbound migrate profile contains locations, a worklist or CDL is generated for each location or DB2 subsystem that is specified.
For example, if your installation supports systems in New York, Dallas, and Atlanta, you can create a profile with a location for each system. Each location can have change rules that tailor object names and attributes for that system. Each location can have its own change rules, or can reference the change rules of another location within the profile.

You can also group several locations under a single Group ID to generate just one worklist or one CDL file. Using the Group ID enables you to execute all of the work for a single subsystem in one process. This action is useful when you are migrating or updating several clones of a structure that reside on the same subsystem.

For example, if a development facility has 25 developers, each of whom have their own copy of the application schema, you could use a Group ID to migrate all 25 copies of the schema within a single worklist.

**CDL files**

CDL is a BMC proprietary language that you use to specify changes to DB2 data structures. You can use CDL files to transmit data structure changes between subsystems or to provide a record of changes to data structures.

CDL files specify the changes that are required to transform a primary set of data structures to a secondary set of data structures. You use CDL files in the change, import, and recovery processes.

**CHANGE MANAGER DDL files**

CHANGE MANAGER can generate executable SQL DDL statements (in SPUFI format) when generating a baseline report.

DDL files are used as input by the Import, Baseline, and Compare components.

**CHANGE MANAGER internal tables**

CHANGE MANAGER uses several types of internal tables to perform tasks.

The tables that CHANGE MANAGER uses are as follows:

- BL tables, which store information about baselines (data for full-recovery baselines is stored in external data sets)
- CD tables, which store change and migration specifications for each work ID
- CM tables, which store work ID, change rule, and sync data
CP tables, which store information about CM/PILOT task IDs, applications, scripts, and DML.

Access to most of the internal tables is controlled by the granting of EXECUTE authority on the appropriate plans.

**Task IDs**

A task ID is a unit of work in the CM/PILOT component.

Each task ID has a unique name and contains information that you provide through the panel to perform a CHANGE MANAGER process. The information that you provide is based on the script you select for the task ID. All work in CM/PILOT is accomplished by processing task IDs.

**Scripts**

The panels that the CM/PILOT component displays are based on predefined steps called scripts.

Each script contains ordered steps that prompt you for the information that is required to perform a change management task. CHANGE MANAGER provides you with several scripts; you can also create your own script, or copy, change, and rename any of the scripts that are supplied by CHANGE MANAGER.

**Applications**

In the CM/PILOT component, you can associate a group of CHANGE MANAGER profiles that are used repeatedly for the change management tasks of a specific DB2 application.

This association is called an application.

**DML**

In the CM/PILOT component, you can use DML statements to update, delete, and migrate data structures.

You can change or migrate multiple objects in a single DML statement. The panels for two of the CM/PILOT scripts help you create the DML statements.

**CM/PILOT worklists**

A CM/PILOT worklist is a data set that contains the ordered commands, keywords, and parameters that are needed to process a task ID.
The format of the CM/PILOT worklist is similar to an Analysis worklist. The JCL that processes the CM/PILOT worklist contains the names of the output data sets that an Analysis worklist uses and the associated Execution JCL. When a CM/PILOT worklist is successfully processed, the result is the creation of an Analysis worklist and the Execution JCL to process the worklist.

Solution integration

ALTER and CHANGE MANAGER are also components of BMC solutions.

ALTER is a component of the Administrative Assistant solution. CHANGE MANAGER is a component of the Database Administration, and the BMC Object Administration for DB2 solutions.

Administrative Assistant for DB2

The Administrative Assistant solution enables users of all experience levels to navigate through the DB2 catalog quickly and to manage a complex DB2 environment easily.

For more information, see these documents:

■ BMC Administrative Assistant for DB2 release notes
■ Installation System Reference Manual
■ Installation System Quick Start
■ BMC Products and Solutions for DB2 Customization Guide

Database Administration for DB2

You can use the Database Administration solution to manage your DB2 databases quickly, efficiently, and effectively.

For more information, see these documents:

■ Database Administration for DB2 release notes
■ Installation System Reference Manual
BMC Object Administration

You can use the BMC Object Administration for DB2 solution to manage your DB2 databases quickly, efficiently, and effectively.

For more information, see these documents:

- BMC Object Administration for DB2 release notes
- Installation System Reference Manual
- Installation System Quick Start
- BMC Products and Solutions for DB2 Customization Guide
Overview of the ALTER and CHANGE MANAGER interface

ALTER and CHANGE MANAGER are designed to help you and your organization manage data structures within a DB2 subsystem or between subsystems.

This section introduces you to the interface for ALTER and CHANGE MANAGER, then steps you through the tasks that are necessary to create and migrate DB2 objects.

Before you use the product

Before you use ALTER and CHANGE MANAGER, ensure that you have completed the tasks necessary to install the products.

For information, see:

- Installation System Reference Manual
- Installation System Quick Start
- BMC Products and Solutions for DB2 Customization Guide

ALTER and CHANGE MANAGER interface

The ALTER and CHANGE MANAGER ISPF user interface consists of dialog panels that comply with Common User Access (CUA) conventions.

To call or start ALTER or CHANGE MANAGER, you can use the BMC Software Administrative Products for DB2 panel that BMC provides (see Figure 4 on page 48).
If you have CHANGE MANAGER installed, ALTER is not an option on this panel.

Figure 4: Administrative products for DB2 panel

The BMC Software Administrative Products for DB2 panel contains the following features:

- The COMMAND line, on which you enter TSO commands
- The product selection list, which includes an input field with which you can specify an option from the list by entering a number
- The DB2 SSID field, in which you specify the DB2 subsystem (SSID) to be used
- The DB2 Catalog Access field, in which you specify whether to use the DB2 catalog data directly or to use a copy or a view of the DB2 catalog
- The Use Shared or Individual product ISPF APPLID field, in which you specify whether to use a shared or individual ISPF profile ID

If you want to specify an ISPF Application ID (APPLID) for each product, use an individual APPLID. If you want to specify an APPLID for each DB2 subsystem, use a shared APPLID.

Main Menu

After you select ALTER or CHANGE MANAGER from the Administrative Products for DB2 panel, the product’s Main Menu is displayed.

You can use the ALTER Main Menu to select options to process work IDs, change installation options, perform specific tasks, and view a list of the latest enhancements to the product.
You can use the CHANGE MANAGER Main Menu (see Figure 5 on page 49) to perform the same functions as ALTER, in addition to building CDL, processing baselines, using baseline and migrate profiles, and accessing the CM/PILOT component of CHANGE MANAGER.

**Figure 5: CHANGE MANAGER Main Menu**

<table>
<thead>
<tr>
<th>Command</th>
<th>Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACMFMAIN</td>
<td>CHANGE MANAGER 11.02.00 Main Menu</td>
</tr>
<tr>
<td>Command ====&gt;</td>
<td></td>
</tr>
</tbody>
</table>

- Select an option. Then press Enter.
  - 1. WORKID (Import, Specify, Analyze, Execute)
  - 2. Compare (CDL Build)
  - 3. Baselines
  - 4. Baseline Profiles
  - 5. Migrate Profiles
  - 6. Environment
  - 7. Options
  - 8. Task List
  - 9. CM/PILOT

- ? About This Release

Tip: To view a long name value, position the cursor on the value and press PF4 (ZOOM).

Commands: HELP END

(c) Copyright 1992-2015 BMC Software, Inc

---

**Displaying the current product environment**

From the Main Menu of ALTER or CHANGE MANAGER, you can list helpful information about the product, data set names, plan names, and applied fixes.

**To display information about the environment**

1. To display information about the current ALTER or CHANGE MANAGER environment, type `ENVI` in the **Command** line on the Main Menu and press **Enter**.
The BMC Environment panel is displayed, as shown in Figure 6 on page 50.

**Figure 6: BMC Environment panel**

```
DEGA ----------------------------- BMC ENVIRONMENT ----------------------------
Command ===>                                                  SCROLL ===> PAGE
Product . . . . . . . : CHANGE MANAGER 11.02.00               Commands: MAINT
Distribution Date . . : 05/06/2015                        More:  +
DB2 Version . . . . . : 10.01.5 IS IN NEW FUNCTION MODE
ISPF APPLID . . . . . : DE12
DOPTs Names : DMB2QEGA      DASD MGR : DSB2QDOP      CATALOG MGR : DC112DOP
Initial POF DSN: AUS.DOPSEC.CNTL(JX82QEGA)
User POF DSN   : AUS.DOPSEC.CNTL(JX82QEHJ)

Lines 1 to 12 of 12
Act    Type   Object                                MORE:
************************************** TOP ***********************************
PL     ACMB2QDE  Environment Plan
PL     ACMB2QDF  Front End Plan
PL     ACMB2QDS  Specification Plan
PL     ACMB2QDA  Analysis Plan
PL     ACMB2QDI  Change Management Import Plan
PL     ACMB2QDC  Change Management Compare Plan
PL     ACMB2QDB  Change Management Baseline Plan
PL     ACMB2QDR  Change Management Report Plan
PL     ACMB2QXM  Execution Public Plan
PL     ACMB2QXA  Execution Auth Plan
PL     ACMB2QDL  LOB Loader Plan
PL     TISINSTL  Execution Inst Auth Plan
************************************ BOTTOM **********************************
```

2. To obtain a list of the synonyms and tables that are dependent upon each plan, type SY in the Act column adjacent to the name of the plan.

3. To display information about fixes (PTFs) that you have applied to the product, type MAINT in the Command line on the BMC Environment panel and press Enter.

The Product Maintenance List panel is displayed.

**Using Fast Path Navigation**

The Installation System for the Administrative products provides a feature called Fast Path Navigation. This feature enables you to switch from one product to another, and then return to the original product.

**To initiate Fast Path Navigation**

1. On the Command line of the current product, enter the command corresponding to the product to which you want to switch (see Table 1 on page 51).
Table 1: Fast Path Navigation Commands

<table>
<thead>
<tr>
<th>Product</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER</td>
<td>BMCALTER</td>
</tr>
<tr>
<td>CHANGE MANAGER</td>
<td>BMCCHG</td>
</tr>
<tr>
<td>CATALOG MANAGER</td>
<td>BMCCAT</td>
</tr>
<tr>
<td>DASD MANAGER PLUS for DB2</td>
<td>BMCDASD</td>
</tr>
</tbody>
</table>

For example, if you are currently using ALTER and want to view an object description in CATALOG MANAGER, enter BMCCAT on the ALTER Command line. The main menu for the requested product is displayed. In this case, the ALTER session is temporarily suspended and then resumed when you exit CATALOG MANAGER.

For more information about enabling Fast Path Navigation, see:

- *Installation System Reference Manual*
- *Installation System Quick Start*
- *BMC Products and Solutions for DB2 Customization Guide*

**ISPF commands**

Most of the ISPF commands in the product operate in the same manner as they do in other ISPF applications.

Table 2 on page 51 describes the most commonly used ISPF commands.

Table 2: ISPF commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CANCEL</td>
<td>Returns to the previous panel without saving any change that you made on the current panel</td>
</tr>
</tbody>
</table>
| DOWN (or F8) | Scrolls the panel down  
More: + on a panel indicates that more information is available below the current line.  
Scrolling is available on a Model 2 3270 mainframe terminal, which uses a 24-line by 80-column display. |
| END (or F3) | Validates and processes information, the same as the Enter key  
In some panels, pressing END returns to the previous panel. |
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENTER</td>
<td>Processes information that is typed on the panel and executes any specified commands. For a sequence of related panels, pressing Enter validates the information on the current panel and displays the next panel in the sequence.</td>
</tr>
<tr>
<td>HELP (or F1)</td>
<td>Provides panel-level Help</td>
</tr>
<tr>
<td>LEFT (or F10)</td>
<td>Scrolls the panel to the left. More: &lt; on a panel indicates that more information is available to the left.</td>
</tr>
<tr>
<td>NEXT</td>
<td>Displays the next panel in a sequence</td>
</tr>
<tr>
<td>PFSHOW</td>
<td>Displays the active function keys. Some panels use every available line to display input variables. To display all variables, enter PFSHOW OFF on the Command line.</td>
</tr>
<tr>
<td>PREV (or F12)</td>
<td>Displays the previous panel in a sequence</td>
</tr>
<tr>
<td>RIGHT (or F11)</td>
<td>Scrolls the panel to the right. More: &gt; on a panel indicates that more information is available to the right.</td>
</tr>
<tr>
<td>SPLIT (or F2)</td>
<td>Divides the panel and displays the ISPF Primary Option Menu in the new panel. If you start the product on both panels, ensure that each product is at the same version, release, and maintenance level.</td>
</tr>
<tr>
<td>SSE</td>
<td>Starts the BMC Simple Space Estimation (SSE) feature to estimate space requirements for table space or index objects</td>
</tr>
<tr>
<td>SWAP (or F9)</td>
<td>Switches from one split panel to another</td>
</tr>
<tr>
<td>UP (or F7)</td>
<td>Scrolls the panel up. More: - on a panel indicates that more information is available above the current line. Scrolling is available on a Model 2 3270 mainframe terminal, which uses a 24-line by 80-column display.</td>
</tr>
</tbody>
</table>
### Command Description

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZOOM (or F4)</td>
<td>Displays the full value of an object with a long name in either a dialog or a panel. The cursor must be positioned on the long name value. On the following panels, you cannot use the ZOOM key to display the entire name of an object:</td>
</tr>
<tr>
<td></td>
<td>- Scope Rules panel (ACMFMPSL)</td>
</tr>
<tr>
<td></td>
<td>- Change Rules panel (ALUFULL)</td>
</tr>
<tr>
<td></td>
<td>- Help panels</td>
</tr>
<tr>
<td></td>
<td>For example, when you convert a table space from table-controlled partitioning to index-controlled partitioning, the products might display a Help panel that lists up to three partitioned indexes. If those indexes have long names, the product displays a portion of the names and disables the ZOOM key.</td>
</tr>
</tbody>
</table>

---

**BMC message information**

The BMC Administrative products include informational, warning, and error messages.

Most messages are assigned a message identifier (*messageID*) that consists of an alphabetic prefix and a number. Some message identifiers also include a severity code, such as W, E, or I.

**Note**

Messages that are not assigned a message identifier display in the upper right-hand corner of the panel.

---

**Achieving your goals with ALTER and CHANGE MANAGER**

Before products like ALTER or CHANGE MANAGER were developed, changing the data structures in a DB2 environment was, at best, difficult.

The process was also complex, tedious, time consuming, unreliable, and error prone. Whether working in a single DB2 subsystem or in a large, multiple-DB2 subsystem...
environment, the DBA must manage a large number of objects with complex interrelationships.

This section describes how ALTER and CHANGE MANAGER can automate and simplify the goals of changing data structures and migrating structure changes.

Creating, altering, or dropping DB2 objects

Creating, altering, or dropping DB2 objects in ALTER or CHANGE MANAGER is a three-stage process: specification, analysis, and execution.

1 Specification
   Specify the changes that you want to make to the objects.

2 Analysis
   Analyze the specified changes for accuracy and completeness. The result of the analysis stage is a worklist that contains the SQL and the information that is necessary to change the data structures and preserve all dependent objects.

3 Execution
   Execute the worklist that is generated during the analysis stage.

   **Note**
   The changes that you specify in the specification stage and analyze in the analysis stage are actually made to the objects in the execution stage.

For more information, view the Quick Course "CHANGE MANAGER for DB2 - Performing Specification."

The following example illustrates these stages by showing you how to use the Task List feature of CHANGE MANAGER to quickly and easily change the length of a column in a table in your database.

Before you begin a set of tasks to achieve a particular goal, you should allocate your JCL data sets for analysis and execution, ensure that you are using the correct DB2 subsystem, and define your job card with the correct account information.

To modify objects

1 Start the Task List feature.

   The Task List feature of CHANGE MANAGER enables you to achieve various goals by taking you step-by-step through a series of tasks.
On the CHANGE MANAGER Main Menu (see Figure 5 on page 49), select Task List and press Enter.

The Task List Menu is displayed, as shown in Figure 7 on page 55.

Figure 7: Task List Menu

ALUFTASK ------------------------- Task List Menu -----------------------------
Command ===> 

Select a task. Then press Enter.
1. Create, alter, or drop DB2 objects
2. Migrate DB2 objects using Specification
3. Migrate DB2 objects using scope rules from a Migrate Profile

Commands: HELP END

On the Task List Menu, select Create, alter, or drop DB2 objects and press Enter.

The Create, Alter, or Drop DB2 Objects panel is displayed, as shown in Figure 8 on page 55.

Figure 8: Create, Alter, or Drop DB2 Objects panel

ALUFALT -------------------- Create, Alter, or Drop DB2 Objects ----------------
Command ===> 

Specify a change WORKID.
If the WORKID does not exist, the Create WORKID panel will be displayed.
To see a list of WORKIDs, type a wildcard pattern. (For example BM*.*)
Alter WORKID:    RDACRJ.ALTTB

Type S to select processing options. Then press Enter to continue.
- Create or edit a WORKID
- Specify changes to DB2 objects
- Analyze the changes to create a worklist
- Execute the worklist to process the changes
Commands: HELP END

Specify an alter-type work ID.

A work ID enables you to identify, control, and track a unit of work. The alter-type work ID is used to store the changes to the DB2 objects that you specify.

On the Create, Alter, or Drop DB2 Objects panel, type the name of an alter-type work ID, and type S to select all of the processing options. Then, press Enter.
The Create WORKID panel is displayed, as shown in Figure 9 on page 56.

**Figure 9: Create WORKID panel**

<table>
<thead>
<tr>
<th>Command</th>
<th>WORKID NOT FOUND</th>
</tr>
</thead>
<tbody>
<tr>
<td>WORKID</td>
<td>RDACRJ.ALTTB</td>
</tr>
<tr>
<td>Type</td>
<td>1. Alter</td>
</tr>
<tr>
<td></td>
<td>2. Migrate</td>
</tr>
<tr>
<td></td>
<td>3. Receive</td>
</tr>
<tr>
<td>Comment</td>
<td></td>
</tr>
</tbody>
</table>

Select additional panels to display for Migrate type WORKID. Then press Enter.

- WORKID Migrate Options
- WORKID Change Rules

Commands: HELP END CANCEL

b On the Create WORKID panel, select **Alter** for the **Type**. Then, press **Enter**.

The Object Specification panel is displayed, as shown in Figure 10 on page 56.

**Figure 10: Object Specification panel**

<table>
<thead>
<tr>
<th>Command</th>
<th>WORKID ADDED</th>
</tr>
</thead>
<tbody>
<tr>
<td>WORKID</td>
<td>RDACRJ.ALTTB</td>
</tr>
<tr>
<td></td>
<td>_ Changed Objects List</td>
</tr>
<tr>
<td>Specify the object type(s) to be included in a list.</td>
<td></td>
</tr>
<tr>
<td>Stogroup.</td>
<td>(SG Name)</td>
</tr>
<tr>
<td>Database.</td>
<td>(DB Name)</td>
</tr>
<tr>
<td>Tablespace</td>
<td>(DB.TS Name)</td>
</tr>
<tr>
<td>Table</td>
<td>(Own.TB Name)</td>
</tr>
<tr>
<td>Check Cnst.</td>
<td>(Own.TB.Chk)</td>
</tr>
<tr>
<td>Foreign Key</td>
<td>(Own.TB.Rel)</td>
</tr>
<tr>
<td>Index</td>
<td>(Own.IX Name)</td>
</tr>
<tr>
<td>Unique Cnst</td>
<td>(Own.TB.Uniq)</td>
</tr>
<tr>
<td>View</td>
<td>(Own.VW Name)</td>
</tr>
<tr>
<td>Trigger</td>
<td>(Sch.TR Name)</td>
</tr>
<tr>
<td>Alias</td>
<td>(Own.AL Name)</td>
</tr>
<tr>
<td>Synonym</td>
<td>(Own.SY Name)</td>
</tr>
</tbody>
</table>

The following objects are retrieved independently of all other DB2 objects.

- Sequence | (Sch.SQ Name) |
- Stored Proc | (Sch.SP Name)

3 Specify the changes that you want to make to the objects.

The current values of the objects are displayed on panels. You can modify the attributes for objects by specifying new values or by typing new values over the existing values. All of your changes are stored in DB2 tables.

a On the Object Specification panel, type the name of your database, table space, and table. Then, press **Enter**.
The Mixed List panel is displayed, as shown in Figure 11 on page 57. From this panel, you begin the process of specifying changes to the objects that are listed.

**Figure 11: Mixed List panel**

```
ALUSMXL ER --------------------------- Mixed List ---------------------------
Command ===> Scroll. . PAGE
WORKID . . . : RDACRJ.ALTTB
Commands: CANCEL
Type action next to object and press Enter.
E=Edit  L=Like  D=Drop  U=Undo
Objects 1 to 13 of 52
More: +
```

```
Act Object-Type Objects
************************************ TOP **************************************
DB . . . . . . CJBASIC
TSG. . . . CJBASIC TRT10AIM
TB. . . . CJBASIC T_T10AIM
IX. . . . CJBASIC I_T10AIM1
TSI. . . . CJBASIC T01AICP
TB. . . . CJBASIC T_T01AICP
IXC. . . CJBASIC I_T01AICP1
#TR . . . CJBASIC T011ICP1
#TR . . . CJBASIC T011ICP2
TS . . . . . CJBASIC T02ASSEG
TB. . . . CJBASIC T02ASSEG
IX . . . CJBASIC I_T02ASSEG_P
IX. . . CJBASIC I_T02ASSEG_X
```

b On the Mixed List panel, in the Act field type E next to the table that you want to edit, and press Enter.

The Table Detail panel is displayed, as shown in Figure 12 on page 57.

**Figure 12: Table Detail panel**

```
ALUSOTB ER -------------------------- Table Detail --------------------------
Command ===> Scroll. . PAGE
WORKID . . . : RDACRJ.ALTTB
Commands: HELP END CANCEL PF4=ZOOM
New Values
TB Owner : CJBASIC
TB Name : T_T01AICP
More: +
Database Name . . . : CJBASIC . . .
Tablespace Name : T01AICP . . .
Audit . . . . . : NONE . . .
(ANONE,ALL,CHANGES)
Validproc . . . . .
- Convert to Table Controlled Partitioning
- Convert to Materialized Query Table
Select additional panels to display then press Enter.
- Table Column List
  - Primary Key Columns
  - Partitioning Key Columns
  - MQT Parameters
  - History Table Information
- Auxiliary Objects List
- Parts List and LimitKey Values
- Implicit TS Dssize
- Archive Table Information
```

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface
c On the Table Detail panel, type S to select the Table Column List, and press Enter.

The Table Columns List panel is displayed, as shown in Figure 13 on page 58. This panel displays the current column attributes for the table. In this example, you will change the length of a column.

**Figure 13: Table Columns List panel**

<table>
<thead>
<tr>
<th>Act</th>
<th>Column Name</th>
<th>Length</th>
<th>N</th>
<th>D</th>
<th>U</th>
<th>Type</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>COLC_1</td>
<td>20</td>
<td>N</td>
<td>Y</td>
<td></td>
<td>CHAR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COLSI_2</td>
<td></td>
<td>N</td>
<td>Y</td>
<td></td>
<td>SMALLINT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COLIN_3</td>
<td></td>
<td>N</td>
<td>Y</td>
<td></td>
<td>INTEGER</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COLSI_4</td>
<td></td>
<td>N</td>
<td>Y</td>
<td></td>
<td>SMALLINT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COLDC_5</td>
<td>6.2</td>
<td>N</td>
<td>Y</td>
<td></td>
<td>DECIMAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COLDC_6</td>
<td>9</td>
<td>N</td>
<td>Y</td>
<td></td>
<td>DECIMAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COLDC_7</td>
<td>5.2</td>
<td>N</td>
<td>Y</td>
<td></td>
<td>DECIMAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COLVC_8</td>
<td>15</td>
<td>N</td>
<td>Y</td>
<td></td>
<td>VARCHAR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COLF1_9</td>
<td>21</td>
<td>N</td>
<td>Y</td>
<td></td>
<td>FLOAT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>COLF2_10</td>
<td>53</td>
<td>N</td>
<td>Y</td>
<td></td>
<td>FLOAT</td>
<td></td>
</tr>
</tbody>
</table>

d On the Table Columns List panel, tab to the Length field for the column that you are modifying. Type a new value for the length of the column over the existing value. Press Enter.

e Press END to return to the Table Detail panel. Press END again to return to the Mixed List Panel.
The Mixed List panel, shown in Figure 14 on page 59, displays an "*A" next to the table, indicating that you have made a change to the table.

**Figure 14: Mixed List panel showing changes**

<table>
<thead>
<tr>
<th>Command</th>
<th>WORKID</th>
<th>Type action next to object and press Enter.</th>
<th>More:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>E=Edit  L=Like  D=Drop  U=Undo</td>
<td>+</td>
</tr>
<tr>
<td>Scroll. . PAGE</td>
<td>RDACRJ.ALTTB</td>
<td>Objects 1 to 13 of 52</td>
<td>+</td>
</tr>
</tbody>
</table>

Objects 1 to 13 of 52

**Figure 15: Analysis Alter Worklist Interface panel**

<table>
<thead>
<tr>
<th>Command</th>
<th>WORKID</th>
<th>Analyze <em>Alter</em> WORKID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Select type of analysis.</td>
</tr>
<tr>
<td>1 1.</td>
<td>1 1. Foreground</td>
<td>2. Batch</td>
</tr>
<tr>
<td>2. Generate a worklist which will convert and reload current data.</td>
<td>2. Generate a worklist which will convert and reload current data and establish a full recovery baseline.</td>
<td></td>
</tr>
<tr>
<td>3. Generate a worklist which will reload data from a previous full recovery baseline.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Press END to return to the Object Specification panel. Press END again.

The Analysis Alter Worklist Interface panel is displayed, as shown in Figure 15 on page 59.

4 Analyze the changes that you specified.

The Analysis component of the product examines the changes that you specified to determine the effect that the changes have on the structure and the related or dependent objects.
On the Analysis Alter Worklist Interface panel, select the option to **Generate a worklist which will convert and reload current data**. Then select to run the analysis in **Foreground**. Press **Enter**.

**Note**
When you run Analysis in foreground, your TSO session will be unavailable until the processing has completed.

The Analysis JCL Processing Interface panel is displayed, as shown in **Figure 16 on page 60**.

**Figure 16: Analysis JCL Processing Interface panel**

```
DEHJ -------------------- Analysis JCL Processing Interface -------------------
Command   ===> WORKID . . . . : RDACRJ.ALTTB

Specify Dataset Names
  Worklist . . . . 'ACM_DB2V10.V102.WLBASE01(ALTTB)'
  Diagnostics . . . SYSDUMP
  Sysout Class . . . A (Used only when Diagnostics = SYSDUMP)

Select foreground processing options. Then press Enter.
  Override
  S Create analysis input
  S Edit analysis input
  R Run analysis
  E Edit Worklist
Commands: BROWSE PREVIOUS HELP END
```

On the Analysis JCL Processing Interface panel, type the name of your **Worklist**, and type **S** to select your processing options. Press **Enter**.

The input for the Analysis process is displayed, as shown in **Figure 17 on page 61**. The input includes the ALUIN input stream, which provides keywords for...
analyzing the changes. For more information about the input stream, see the
*ALTER and CHANGE MANAGER for DB2 Reference Manual*.

**Figure 17: Analysis input**

```plaintext
ISREDDE2   SYS13015.T123932.RA000.RDACRJ.WLIN.H03
Command ===>
Scroll ====> PAGE

****** ******************************************* Top of Data *******************************************
==MSG> -Warning- The UNDO command is not available until you change
==MSG> your edit profile using the command RECOVERY ON.

000001 SSID DEGA
000002 WORKID RDACRJ.ALTTB
000003 SORTDEVT SYSDA
000004 INCLUDE (DATA AMS SQL REBIND )
000005 NOREGENIDENTITY
000006 SMSINCLUDE ()
000007 NOBRMLIB ORDERBY
000008 NOVALPROP
000009 UNLOADEMPTY NOSTOPCOMMIT TABLEACCESS
000010 NOREORGALT
000011 IBMCHECK BMCREORG NOREORGONLINE DYNREORG REORGALL REORGREF
000012 BMCREBUILD
000013 SYNCPOINT 10
000014 STANDALONESTATS DB2STATSUPD TABLEALL KEYCARD NOHISTORY
000015 BMCCOPY BMCLOAD BMCUNLOAD
000016 BMCFASTLOAD
000017 NONULDCOLL
000018 UNLOADTABLE
000019 MAXSYSUT 20
000020 DYNUNLD
000021 DYNCOPY
000022 NOPARTCOPY
000023 COPYDNN(COPY01 )
000024 NOTILCOPY
000025 NOPARALLEL
000026 CLONEDATA

****** ******************************************* Bottom of Data *******************************************
```

c Press **F8** to scroll down the input. Press **END**.

d Press **Enter** to run Analysis.

The results of Analysis are a diagnostic output file (ALUPRINT) and a
worklist.

e To review the results of Analysis (ALUPRINT), you can split your ISPF session
or you can use your normal method to review SYSOUT. The last message in
the diagnostic output (Figure 18 on page 62) indicates that the worklist completed successfully.

**Figure 18: Analysis diagnostic output**

```
SDSF OUTPUT DISPLAY RDACRJ TSU22735 DSID 112 LINE 0 COLUMNS 02 81
COMMAND INPUT ===> SCROLL ===> PAGE
*******************************************************************************
SSID DEGA
WORKID RDACRJ.ALTTB
SORTDEV SYSDA
INCLUDE (DATA AMS SQL REBIND )
NOREGENIDENTITY
SMSINCLUDE ()
NODBRMLIB ORDERBY
NOVVALPROP
UNLOADEMPTY NOSTOPCOMMITE TABLEACCESS
NOREORGALT
IBMCHECK BMCREORG NOREORGNONE DB2STATSUPD TABLEALL KEYCARD NOHISTORY
BMCCOPY BMCMLOAD BMCUNLOAD
BMCFASTLOAD
NOUNLDLOG
UNLOADTABLE
MAXSYSUT 20
DYNUNLD
DYNCOPY
NOPARTCOPY
COPYDDN(COPY01 )
NOUTILCOPY
NOPARALLEL
CLONEDATA
*BMC396243I DB2 VERSION 10 NFM
*** PIFS APPLIED TO PRODUCT:
*** NONE
*******************************************************************************
*** WORKLIST GENERATION 2013-01-15 12:42:46.259803
*** CHANGE MANAGER FOR DB2 VERSION 11.01.00(06/25/2013)
WORKLIST GENERATION EXECUTION FOR WORKID RDACRJ.ALTTB ON SSID DEGA
*******************************************************************************
LIMIT KEY VALIDATION STARTING, COMPILED Dec 20 2012 AT 21:39:04
LIMIT KEY VALIDATION COMPLETE
*BMC396243I DB2 VERSION 10 NFM
WORKLIST WAS WRITTEN TO DATASET 'ACM.DB2V10.V102.WLBASE01(ALTTB)
WORKLIST GENERATION COMPLETE WORKID RDACRJ.ALTTB
*******************************************************************************
```

Figure 19 on page 62 shows the worklist that was written to the data set that you specified on the Analysis JCL Processing Interface panel.

**Figure 19: CHANGE MANAGER alter-type worklist**

```
*** CHANGE MANAGER FOR DB2 VERSION 11.01.00(06/25/2013)
*******************************************************************************
*** PIFS APPLIED TO PRODUCT:
*** NONE
*******************************************************************************
* ALUIN PARMS: *
* SSID DEGA *
* WORKID RDACRJ.ALTTB *
* SORTDEV SYSDA *
* INCLUDE (DATA AMS SQL REBIND ) *
* NOREGENIDENTITY *
* SMSINCLUDE () *
* NODBRMLIB ORDERBY *
* NOVVALPROP *
* UNLOADEMPTY NOSTOPCOMMITE TABLEACCESS *
* NOREORGALT *
```
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  63
f Press END.

The Execution JCL Build Interface panel is displayed, as shown in Figure 20 on page 64.

**Figure 20: Execution JCL Build Interface panel**

ACMFEXCO ---------------- Execution JCL Build Interface ----------------
Command ===>

WORKID . . . . : RDACRJ.ALTTB

Select JCL and run type. Then press Enter.

<table>
<thead>
<tr>
<th>JCL Type</th>
<th>1 1. Build Initial JCL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run Type</td>
<td>1 1. Build JCL in Foreground</td>
</tr>
</tbody>
</table>

Commands: NEXT HELP END

5 Execute the changes that you specified and analyzed.

The Execution component of the product uses the worklist as its input job stream and performs the tasks contained in the worklist. The worklist acts as a detailed procedure for implementing the changes that you defined in the specification stage.

a On the Execution JCL Build Interface panel, press **Enter** to accept the defaults.
The Execution Pre- and Post-Processing Interface panel is displayed, as shown in Figure 21 on page 65.

Figure 21: Execution Pre- and Post-Processing Interface panel

```
ACMFEXC1 ---------- Execution Pre- and Post-Processing Interface --------------
Command ===>  
WORKID . . . . : RDACRJ.ALTTB
Specify additional steps to be included in this job. Then press Enter.
Pre-Execution Compare
1. Do not do a compare before execution
2. Compare two previous baselines
3. Compare a previous baseline to the current catalog
Pre-Execution Baseline
1. Do not build a baseline before execution
2. Build a baseline before execution
Post-Execution Compare
1. Do not do a compare after execution
2. Compare a previous baseline to the new catalog structures
3. Compare the baseline built during a previous job step to the new catalog structures
4. Create CDL to fall back to a previous baseline
Post-Execution Baseline
1. Do not build a baseline after execution
2. Build a baseline after execution
Commands: HELP PREVIOUS END
```

b On the Execution Pre- and Post-Processing Interface Panel, press Enter to accept the defaults.

The Execution JCL Processing Interface panel is displayed, as shown in Figure 22 on page 65.

Figure 22: Execution JCL Processing Interface panel

```
ACMFEXC3 --------------- Execution JCL Processing Interface -------------------
Command ===>  
WORKID . . . . : RDACRJ.ALTTB
Specify Dataset Names.
Execution JCL . 'ACM.DB2V10.V102.EXECJCL(ALTTB)'
Worklist . 'ACM.DB2V10.V102.WLBASE01(ALTTB)'
Diagnostics . . . . SYSOUT
JCL Build Options.
Dataset Sizing : No Sizing        (use Override Defaults to set or change)
Select Processing Options. Then press Enter to continue.
Override Defaults
  Edit Worklist
S Build Execution JCL
S Edit Execution JCL
S Submit Execution JCL

Commands:  BROWSE VIEW PREVIOUS HELP END
```

c On the Execution JCL Processing Interface panel, specify the data set name for the Execution JCL, accept the defaults for the JCL build options, and type S to select the processing options. Press Enter.

The Execution JCL is displayed, as shown in Figure 23 on page 66. The JCL includes the //AEXIN DD statement (input stream), which includes keywords
that are used in executing the JCL. For more information about the input stream, see the ALTER and CHANGE MANAGER for DB2 Reference Manual.

**Figure 23: Execution JCL for an alter-type worklist**

```
//RDACRJ JOB (5715),'EXECUTE-ALTTB',
// CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1),
// NOTIFY=RDACRJ
/*ROUTE XEQ BMCPXL1
*/JOBPARM SYSAFF=DB2B

/***************************************************************************/
/*  CREATED BY :  RDACRJ */
/*  TIMESTAMP  :  01/15/2013.12.54.25 */
/*  ENVIRONMENT:  ISPF 6.3MVS   TSO */
/*  RELEASE    :  V11.01.00 06/25/2013 */
/*  DB2 VERSION:  10015 */
/***************************************************************************/

S /* BMC CHANGE MANAGER EXECUTION */

/STEP1 EXEC PGM=AEXEMAIN,
// REGION=0M,MEMLIMIT=NOLIMIT,PARM='JMBISEGA',
// COND=(4,LT)
/STEPLIB DD DSN=SYS3.DEGA.DSNEXIT,DISP=SHR
// DSN=CSGI.DB2V10.SDSNLOAD,DISP=SHR
// DSN=AUS.DOPSEC.LOAD

ABNLIGNR DD DUMMY
SYSDUMP DD SYSOUT=* 
SYSTEMR DD SYSOUT=* 
ASUSRPRNT DD SYSOUT=* 
UTPRINT DD SYSOUT=* 

/STATISTICS WORK DD STATEMENTS

STPRIN01 DD SYSOUT=* 
RNPRIN01 DD SYSOUT=* 
SYSOUT DD SYSOUT=* 
AJXERR DD SYSOUT=* 
AEXIN DD *
ACM
SSID DEGA WORKID RDACRJ.ALTTB
DYWORKUNIT SYSDA 
DASDDOPT DSA2QDOP
CATDOPT DC102DOP
STOPWAIT 3
STOPWTSECS 10
LOCK EXCLUSIVE

SYSIN DD SPACE=(CYL,(15,15)),UNIT=SYSDA,DISP=(NEW,DELETE),
// DSORG=PS,RECL=80,BLKSIZ=3200,RECFM=FB
// SYSTSN DD SPACE=(TRK,(1,1)),UNIT=SYSDA,DISP=(NEW,DELETE),
// DSORG=PS,RECL=80,BLKSIZ=3200,RECFM=FB
// SYSPRINT DD SPACE=(CYL,(15,15)),UNIT=SYSDA,
// DISP=(NEW,DELETE)
// SYSTSPRT DD SPACE=(CYL,(15,15)),UNIT=SYSDA,DISP=(NEW,DELETE),
// DSORG=PS,RECL=137,BLKSIZ=4096,RECFM=WBA
// ALUIN DD SPACE=(TRK,(1,1)),UNIT=SYSDA,DISP=(NEW,DELETE),
// DSORG=PS,RECL=80,BLKSIZ=3200,RECFM=FB
// ALUPRINT DD SPACE=(CYL,(15,15)),UNIT=SYSDA,DISP=(NEW,DELETE)
// AJXPRINT DD SPACE=(CYL,(15,15)),UNIT=SYSDA,DISP=(NEW,DELETE)

//SYSPROC DD DSN=AEX.QA1020.BMCCLIB,DISP=SHR
//ISPMLIB DD DSN=AEX.QA1020.BMCMLIB,DISP=SHR
//AEXPRINT DD SYSOUT=* 
//WORKLIST DD DISP=SHR,
// DSN=ACM.DB2V10.VI02.WLBASE01(ALTTB)
```
After you review the JCL, press END to return to the Execution JCL Processing Interface panel.

From the Execution JCL Processing Interface panel, press Enter to submit the JCL and process your changes.

The result of execution is a worklist execution log (AEXPRINT).

To review the results of execution in the worklist execution log (AEXPRINT), you can split your ISPF session or you can use your normal method to review SYSOUT. Figure 24 on page 67 shows the worklist execution log.

The last message in the diagnostic output indicates that the work ID status is complete and that the worklist completed successfully. The worklist implemented the structure changes that you specified.

Figure 24: Execution results for an alter-type worklist
AUTHID SWITCHING OPTION 'N' IS IN EFFECT. -AUTH NOT ALLOWED.

EXECUTION AUTHID IS RDACRJ

CHANGE MANAGER FOR DB2 VERSION 11.01.00(06/25/2013)

WORKID DESCRIPTION:

INITIAL WORKID STATUS FOR RDACRJ.ALTTB IS 'N' (NOT STARTED)

UPDATED WORKID STATUS FOR RDACRJ.ALTTB IS 'S' (STARTED)

PTFS APPLIED TO PRODUCT:

NONE

ALUIN PARMS:

SSID DEGA

WORKID RDACRJ.ALTTB

SORTDEVT SYSDA

INCLUDE (DATA AMS SQL REBIND)

NOREGENIDENTITY

SMSINCLUDE ()

ALUIN PARMS:                          *  
SSID DEGA                             *
WORKID RDACRJ.ALTTB                  *
SORTDEVT SYSDA                       *
INCLUDE (DATA AMS SQL REBIND)         *
NOREGENIDENTITY                      *
SMSINCLUDE ()                        *

BMC EXECUTION FOR DB2

VERSI ON: V11.01.00
(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE,
INC.

2013-01-15 12.55.14 ON SYSID DB2B  WORKLIST EXECUTION

LOG  PAGE  2

STATEMENTS

NODBRMLIB ORDERBY

UNLOADEMPTY NOSTOPCOMMIT TABLEACCESS

IBMCHECK BMCREORG NOREORGONLINE DYNREORG REORGALL REORGREF

BMCREBUILD

SYNCPONT 10

STANDALONESTATS DB2STATSUPD TABLEALL KEYCARD NOHISTORY

BMCCOPY BMCLOAD BMCUNLOAD

BMCFASTLOAD

OUNLD Coll

UNLOADTABLE

MAXSYSUT 20

DYNUNLD

DYNCOPY

NOPARTCOPY

COPYDDN(COPY01)

NOUTILCOPY

NOPARALLEL

CLONEDATA

TIME 000050 '2013-01-15-12.42.46.259803'  24927017

SSID 000100 DEGA  64659863

WKID 000150 RDACRJ.ALTTB  59398639

SYNC 000200 START OF EXECUTION SYNCPONT  52431719

BMC EXECUTION FOR DB2

VERSI ON: V11.01.00
(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE,
INC.

2013-01-15 12.55.14 ON SYSID DB2B  WORKLIST EXECUTION
...
V10R2.00 *****

**** B M C   R E O R G   P L U S   F O R   D B 2

**** B M C   R E O R G   P L U S   F O R   D B 2

***** B M C   R E O R G   P L U S   F O R   D B 2

V10R2.00 *****

(C) COPYRIGHT 1988 - 2013 BMC SOFTWARE, INC.
REORG PLUS TECHNOLOGY IS PROTECTED BY ONE OR MORE OF THE FOLLOWING U.S.
PATENTS: 5,222,235 AND 7,809,762

BMC500011 UTILITY EXECUTION STARTING 1/15/2013 12:55:15 ...
BMC500021 UTILITY ID = 'ALTTBXXXX'. DB2 SUBSYSTEM ID = 'DEGA'. OPTION MODULE = 'ARU $OPTS'.
BMC50471I z/OS 1.13.0,PID=HBB7780,DFSMS FOR Z/OS=1.13.0,DB2=10.1.0
BMC50471I REGION=OM,BELOW 16M=8792K,ABOVE 16M=1417172K,IEFUSI=NO,CPU=3
BMC50471I MEMLIMIT=17592186040320M,AVAILABLE=17592186040314M,MEMLIMIT SET BY:JCL
BMC50471I REORG PLUS FOR DB2--V10.02.00
BMC50471I NO MAINTENANCE TO REPORT
BMC50471I DB2 UTILITIES COMMON CODE--V10.02.00
BMC50471I NO MAINTENANCE TO REPORT
BMC50471I SOLUTION COMMON CODE--V11.01.00
BMC50471I NO MAINTENANCE TO REPORT
BMC50471I BMCSORT ENGINE--V02.03.01
BMC50471I MAINT: BPJ0195 BPJO0198 BPJO0209 BPJO0250 BPJO0263 BPJO0269 BPJO0276
BPJO0279 BPJO0288 BPJO0308 BPJO0343 BPJO0357
BPJO0471I BPJO0361 BPJO0374 BPJO0390 BPJO0404 BPJO0473 BPJO0503 BPJO0521
BPJO0529 BPJO0591
BMC50471I BMC STATS API--V11.01.00
BMC50471I NO MAINTENANCE TO REPORT
BMC50471I EXTENDED BUFFER MANAGER--V06.01.00
BMC50471I NO MAINTENANCE TO REPORT
BMC50471I ACFORTSS=YES
FSTHRESHOLD=0                           RIDMMAXD=1
BMC50471I ALTRFAIL=RCVRPEND
HASHMAX=YES                             RMAPMEM=0
BMC50471I ANALMAX=1000%
ICDDN=(BMCICY,BMCICZ)
BMC50471I ARC=NO
ICTYPE=AUTO
BMC50471I ARCHDDN=SYSARC
IDCACHE=10000
BMC50471I ARCHFORMAT=BMC
INDEFLM=10
BMC50471I ACPORTSS=DEFAULT
FRONTIER=0                             SDUMP=YES
BMC50471I AUXREORG=DEFAULT
INLINEC=YES
BMC50471I AVAILPAGEPC=100
INLOB=YES
BMC50471I BILDMAX=300%
IXONEX=NO
BMC50471I BMCHIST=YES
IXRANDOM=NO
BMC50471I CBUFFS=30
KEEPDICTIONARY=NO
BMC50471I CONDEXEC=NO
LEAFDSLM=200
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
LOB=YES
BMC50471I COPYLVL=PART
LOCKROW=YES
BMC50471I COPYMAX=100%
LOGFINAL=NONE
BMC50471I COPYSUBSET=NO
LOGMEM=0
BMC50471I CYPREFAIL=TERM
LOGSPIL=(20000,10000)
BMC50471I DATACAP=NO
LOGTHRS=0
BMC50471I DDLDDN=DDLIN

Achieving your goals with ALTER and CHANGE MANAGER

70 ALTER and CHANGE MANAGER for DB2 User Guide
Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  71
Achieving your goals with ALTER and CHANGE MANAGER

72  ALTER and CHANGE MANAGER for DB2 User Guide
<table>
<thead>
<tr>
<th>STATEMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC50470I MGMTCLAS = (NONE,NONE)</td>
</tr>
<tr>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>BMC50470I STORCLAS = (NONE,NONE)</td>
</tr>
<tr>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>BMC50470I THRESHLD = 0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>BMC50470I MAXEXTSZ = ((0,K),(0,K))</td>
</tr>
<tr>
<td>(0,K)</td>
</tr>
<tr>
<td>BMC50470I EXPDT =</td>
</tr>
<tr>
<td>BMC50470I RETPD =</td>
</tr>
<tr>
<td>BMC50470I GDGLIMIT = 5</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>BMC50470I GDGEMPTY = NO</td>
</tr>
<tr>
<td>BMC50470I GDGSCRAT = NO</td>
</tr>
<tr>
<td>BMC50470I DDTYPE = REMPFCPY</td>
</tr>
<tr>
<td>LOCBFCPY</td>
</tr>
<tr>
<td>BMC50470I ACTIVE = NO</td>
</tr>
<tr>
<td>BMC50470I IFALLOC = USE</td>
</tr>
<tr>
<td>BMC50470I ALLOC = N/A</td>
</tr>
<tr>
<td>BMC50470I SMS = NO</td>
</tr>
<tr>
<td>BMC50470I SMSUNIT = NO</td>
</tr>
<tr>
<td>BMC50470I SIZEPCT = (100,100)</td>
</tr>
<tr>
<td>(100,100)</td>
</tr>
<tr>
<td>BMC50470I UNIT = (SYSALLDA,SYSALLDA)</td>
</tr>
<tr>
<td>(SYSALLDA,SYSALLDA)</td>
</tr>
<tr>
<td>BMC50470I UNITCNT = (0,0)</td>
</tr>
<tr>
<td>(0,0)</td>
</tr>
<tr>
<td>BMC50470I VOLCNT = (25,25)</td>
</tr>
<tr>
<td>(25,25)</td>
</tr>
<tr>
<td>BMC50470I AVGVOLSP = ((30000,TRK),(30000,TRK))</td>
</tr>
<tr>
<td>(30000,TRK)</td>
</tr>
<tr>
<td>BMC50470I DSNTYPE = (NONE,NONE)</td>
</tr>
<tr>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>BMC50470I DATACLAS = (NONE,NONE)</td>
</tr>
<tr>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>BMC50470I MGMTCLAS = (NONE,NONE)</td>
</tr>
<tr>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>BMC50470I STORCLAS = (NONE,NONE)</td>
</tr>
<tr>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>BMC50470I THRESHLD = 0</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>BMC50470I MAXEXTSZ = ((0,K),(0,K))</td>
</tr>
<tr>
<td>(0,K)</td>
</tr>
<tr>
<td>BMC50470I EXPDT =</td>
</tr>
<tr>
<td>BMC50470I RETPD =</td>
</tr>
<tr>
<td>BMC50470I GDGLIMIT = 5</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>BMC50470I GDGEMPTY = NO</td>
</tr>
<tr>
<td>BMC50470I GDGSCRAT = NO</td>
</tr>
<tr>
<td>BMC50470I DDTYPE = REMPFCPY</td>
</tr>
<tr>
<td>REMPFCPY</td>
</tr>
</tbody>
</table>
Achieving your goals with ALTER and CHANGE MANAGER

BMC50470I ACTI\[vE = NO
NO
BMC50470I IFALLOC = USE
USE
BMC50470I ALLOC = N/A
N/A
BMC50470I SMS = NO
NO
BMC50470I SMSUNIT = NO
NO
BMC50470I SIZEPCT = (100,100)
(100,100)
BMC50470I UNIT = (SYSALLDA,SYSALLDA)
(SYSALLDA,SYSALLDA)
BMC50470I UNITCNT = (0,0)
(0,0)
BMC50470I VOLCNT = (25,25)
(25,25)
BMC50470I AVGVOLSP = ((30000,TRK),(30000,TRK))
((30000,TRK),(30000,TRK))
BMC50470I DSNTYPE = (NONE,NONE)
(NONE,NONE)
BMC50470I DATACLAS = (NONE,NONE)
(NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)
(NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)
(NONE,NONE)
BMC50470I THRESHLD = 0
0
BMC50470I MAXEXTSZ = ((0,K),(0,K))
((0,K),(0,K))
BMC50470I EXPDT =
BMC50470I RETPD =

**** BMC EXECUTION FOR DB2
**** V E R S I O N: V11.01.00
(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.
2013-01-15 12.55.30 ON SYSID DB2B WORKLIST EXECUTION

LINE PREFIX : DENOTES WORKLIST INPUT

BMC50470I GDGLIMIT = 5
5
BMC50470I GDGEMPTY = NO
NO
BMC50470I GDGSCRAT = NO
NO
BMC50470I DDTYPE = SYSPUNCH
BMC50470I ACTIVE = YES
BMC50470I IFALLOC = USE
BMC50470I ALLOC = N/A
BMC50470I SMS = NO
BMC50470I SMSUNIT = NO
BMC50470I SIZEPCT = (100,100)
BMC50470I UNIT = (SYSALLDA,SYSALLDA)
BMC50470I UNITCNT = (0,0)
BMC50470I VOLCNT = (25,25)
BMC50470I AVGVOLSP = ((30000,TRK),(30000,TRK))
BMC50470I DATACLAS = (NONE,NONE)

74 ALTER and CHANGE MANAGER for DB2 User Guide
Chapter 2 Overview of the ALTER and CHANGE MANAGER interface 75

**BMC EXECUTION FOR DB2**

**** VERSION: V11.01.00

(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 12.55.30 ON SYSID DB2B

WORKLIST EXECUTION

LOG PAGE 8

LINE PREFIX : DENOTES WORKLIST INPUT

<table>
<thead>
<tr>
<th>Statement</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC50470I SMS = NO</td>
<td>NO</td>
</tr>
<tr>
<td>BMC50470I SMSUNIT = NO</td>
<td>NO</td>
</tr>
<tr>
<td>BMC50470I SIZEPCT = (100,100)</td>
<td>(100,100)</td>
</tr>
<tr>
<td>BMC50470I UNIT = (SYSALLDA,SYSALLDA)</td>
<td>(SYSALLDA,SYSALLDA)</td>
</tr>
<tr>
<td>BMC50470I UNITCNT = (0,0)</td>
<td>(0,0)</td>
</tr>
<tr>
<td>BMC50470I VOLCNT = (25,25)</td>
<td>(25,25)</td>
</tr>
<tr>
<td>BMC50470I DDSTYPE = (30000,TRK),(30000,TRK)</td>
<td>(30000,TRK),(30000,TRK)</td>
</tr>
<tr>
<td>BMC50470I DATACLAS = (NONE,NONE)</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>BMC50470I DATACLS = (NONE,NONE)</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>BMC50470I MGMTCLAS = (NONE,NONE)</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>BMC50470I STORCLAS = (NONE,NONE)</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>BMC50470I THRESHLD = 0</td>
<td>0</td>
</tr>
<tr>
<td>BMC50470I MAXEXTSZ = (0,K),(0,K)</td>
<td>(0,K),(0,K)</td>
</tr>
<tr>
<td>BMC50470I EXPDT =</td>
<td></td>
</tr>
<tr>
<td>BMC50470I RETPD =</td>
<td></td>
</tr>
<tr>
<td>BMC50470I GDGLIMIT = 5</td>
<td>5</td>
</tr>
<tr>
<td>BMC50470I GDGEMPTY = NO</td>
<td>NO</td>
</tr>
<tr>
<td>BMC50470I GDGSCRAT = NO</td>
<td>NO</td>
</tr>
<tr>
<td>BMC50470I DDTYPE = REMPLCPY</td>
<td>REMBLCPY</td>
</tr>
<tr>
<td>BMC50470I ACTIVE = NO</td>
<td>NO</td>
</tr>
<tr>
<td>BMC50470I IFALLOC = USE</td>
<td>USE</td>
</tr>
<tr>
<td>BMC50470I ALLOC = N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>
BMC50470I DDGEMPTY = NO
BMC50470I DDGSCRAT = NO
BMC50483I UNLOAD DSNPAT=&UID..BMC..&TSIX..&DDNAME
BMC50483I WORK DSNPAT=&UID..BMC..&TSIX..&DDNAME
BMC50483I SORTWORK DSNPAT=&UID..&UTILPFX..&DDNAME
BMC50483I SYSPUNCH DSNPAT=&UID..&UTILPFX..&DDNAME
BMC50483I LOCPFCPY DSNPAT=&UID..&DDNAME..&TSIX..F&PART..T&TIME
BMC50483I LOCPICPY DSNPAT=&UID..&DDNAME..&TSIX..F&PART..T&TIME
BMC50483I REMPFCPY DSNPAT=&UID..&DDNAME..&TSIX..F&PART..T&TIME
BMC50483I REMPICPY DSNPAT=&UID..&DDNAME..&TSIX..F&PART..T&TIME
BMC50483I REMPFCPY DSNPAT=&UID..&DDNAME..&TSIX..F&PART..T&TIME
BMC50483I REMPICPY DSNPAT=&UID..&DDNAME..&TSIX..F&PART..T&TIME
BMC50483I REMBFCPY DSNPAT=&UID..&DDNAME..&TSIX..F&PART..T&TIME
BMC50483I REMBICPY DSNPAT=&UID..&DDNAME..&TSIX..F&PART..T&TIME
BMC50471I DB2 DSNHDECP MODULE SETTINGS:
BMC50471I VERSION = 1010
BMC50471I SUBSYSTEM DEFAULT = DEGA
BMC50471I CHARACTER SET = ALPHANUM
BMC50471I DATE FORMAT = USA
BMC50471I TIME FORMAT = USA
BMC50471I LOCAL DATE LENGTH = 0
BMC50471I LOCAL TIME LENGTH = 0
BMC50471I DECIMAL POINT = PERIOD
BMC50471I DECIMAL ARITHMETIC = 15
BMC50471I DELIMITER = DEFAULT
BMC50471I SQL DELIMITER = DEFAULT
BMC50471I ENCODING SCHEME = EBCDIC
BMC50471I APPL. ENCODING SCHEME = EBCDIC
BMC50471I MIXED = NO
BMC50471I EBCDIC CCSID = (37,65534,65534)
BMC50471I ASCII CCSID = (819,65534,65534)
BMC50471I UNICODE CCSID = (367,1208,1200)
BMC50028I DB2 MODE = NFM

**** BMC EXECUTION FOR DB2 ****
**** VERSION: V11.01.00 ****
(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 12.55.30 ON SYSID DB2B WORKLIST EXECUTION
LOG PAGE 9

STATEMENTS

BMC50471I BMCUTIL ='BMCUTIL.CMN_BMCUTIL'
BMC50471I BMCSYNC ='BMCUTIL.CMN_BMCSYNC'
BMC50471I BMCHIST = 'BMCUTIL.CMN_BMCHIST'
BMC50471I BMCDICT = 'BMCUTIL.CMN_BMCDICT'
BMC50471I BMCCOPY = 'BMCUTIL.CMN_BMCCOPY'
BMC50501I REORG TABLESPACE CJBASIC.T01AICP
BMC50501I UNLOAD CONTINUE
BMC50501I DELETEDFILES NO
BMC50501I
BMC50501I DDTYPE UNLOAD ACTIVE YES
BMC50501I ALLOC ANY
BMC50501I IFALLOCUSE
BMC50501I UNIT SYSDA
BMC50501I DSNPAT 'RDACRJ.DEGA.ALTTB.SR000001.P&PART'
BMC50501I UNLDDN SYRC
BMC50501I
BMC50501I DDTYPE WORK ACTIVE YES

76 ALTER and CHANGE MANAGER for DB2 User Guide
BMC50102I  ALLOC ANY
BMC50102I  IFALLOC USE
BMC50102I  UNIT  SYSDA
BMC50102I  DSNPAT 'RDACRJ.ALTTB.STEP1.SU000001.&DDNAME'
BMC50102I  WORKDDN DUT
BMC50102I
BMC50102I  DDTYPE LOCPFCPY     ACTIVE YES
BMC50102I      ALLOC ANY
BMC50102I      IFALLOC USE
BMC50102I      UNIT  SYSDA
BMC50102I  DSNPAT 'RDACRJ.ALTTB.T01AICP.P000.LP0001.P&PART'
BMC50102I              COPY YES
BMC50102I              INLINE YES
BMC50102I              COPYLVL PART
BMC50102I                     COPYDDN (SYDL)
BMC50102I              SORTDATA LOG NO
BMC50102I              SORTNUM  32
BMC50102I              SORTDEV SYSDA
BMC50102I              SHRLEVEL REFERENCE
BMC501009I  'DDTYPE UNLOAD' OPTION SPECIFIED.  OPTION 'ALLOC' IGNORED
BMC501009I  'DDTYPE WORK' OPTION SPECIFIED.  OPTION 'ALLOC' IGNORED
BMC501009I  'DDTYPE LOCPFCPY' OPTION SPECIFIED.  OPTION 'ALLOC' IGNORED
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:00
BMC50244I PROCESSING PAGESET DBID(4295) PSID(18) PART(1) THAT MIGHT CONTAIN VERSIONED DATA
BMC50245I REGISTERING HEADER PAGE - 00000000 FOR DBID(4295) PSID(18)
BMC50244I PROCESSING PAGESET DBID(4295) PSID(18) PART(2) THAT MIGHT CONTAIN VERSIONED DATA
BMC50245I REGISTERING HEADER PAGE - 00100000 FOR DBID(4295) PSID(18)
BMC50244I PROCESSING PAGESET DBID(4295) PSID(18) PART(3) THAT MIGHT CONTAIN VERSIONED DATA
BMC50244I PROCESSING PAGESET DBID(4295) PSID(18) PART(4) THAT MIGHT CONTAIN VERSIONED DATA
BMC50245I REGISTERING HEADER PAGE - 00200000 FOR DBID(4295) PSID(18)
BMC50245I REGISTERING HEADER PAGE - 00300000 FOR DBID(4295) PSID(18)

**** B M C   E X E C U T I O N   F O R   D B 2

**** V E R S I O N:  V11.01.00
(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 12.55.30 ON SYSID DB2B

LOG  PAGE 10

STATEMENTS

BMC50102I 4: SAMPLING STATISTICS:
PART=1,TP=3,SP=1,SR=1,AVGR=210,SD=64,SE=0,AVGF=100,SD=64,SE=0,EP=0
BMC51265I ESTIMATED CARDINALITY OF SPACE = 2  AVG SORTWK ROW LENGTH = 210  AVG UNLOAD ROW LENGTH = 210
BMC50484I ESTIMATED CARDINALITY OF PART 0001 = 1  AVG SORTWK ROW LENGTH = 210  AVG UNLOAD ROW LENGTH = 210
BMC50484I ESTIMATED CARDINALITY OF PART 0002 = 0  AVG SORTWK ROW LENGTH = 0  AVG UNLOAD ROW LENGTH = 0
BMC50484I ESTIMATED CARDINALITY OF PART 0003 = 0  AVG SORTWK ROW LENGTH = 0  AVG UNLOAD ROW LENGTH = 0
BMC50484I ESTIMATED CARDINALITY OF PART 0004 = 1  AVG SORTWK ROW LENGTH = 210  AVG UNLOAD ROW LENGTH = 210
BMC51264I UNLOAD WILL READ 2 DATA PAGES FROM SPACE 'CJBASIC.T01AICP'
BMC50041I 0: ZIIP NOT ENABLED (0) USING XBM SUBSYSTEM XBMB
BMC50041I ANALYZE PHASE COMPLETE. ELAPSED TIME = 00:00:00
BMC50041I 0: ZIIP ENABLED (0) USING XBM SUBSYSTEM XBMB
BMC50425I &JOBNAME RDACRJE &STEPNAME STEP1 &DB CJ BASIC &TSIX
T01AICP &RTYPE TS
BMC50425I &UID RDACRJ &DATE 011513 &TIME 125515 &SSID DEGA
DEGA &UTIL ALTTBXXX &UTILSFX &DATE8 01152013 &GRPNM DEGA
DEGA &VCAT DEGACAT
BMC50425I &TIME4 1255 &DATEJ 2013015 &JDATE 13015
BMC50877I POINT OF CONSISTENCY ESTABLISHED AT RBA/LRSN = 00BBD14E7285
BMC50445I REORG PLUS DYNAMIC FILE ALLOCATION REPORT
BMC50446I
UNIT
BMC50447I DDNAME DSNAME DATACLAS MGMTCLAS STORCLAS PRI SEC PRI SEC
SYDL0001 RDACRJ.ALTTB.T01AICP.P000.LP0001.P001 SYSDA 12 1 1 1 TRK
SYDL0002 RDACRJ.ALTTB.T01AICP.P000.LP0001.P002 SYSDA 8 0 1 0 TRK
SYDL0003 RDACRJ.ALTTB.T01AICP.P000.LP0001.P003 SYSDA 8 0 1 0 TRK
SYDL0004 RDACRJ.ALTTB.T01AICP.P000.LP0001.P004 SYSDA 12 1 1 1 TRK
SYRC0001 RDACRJ.DEGA.ALTTB.SR000001.P001 SYSDA 1 1 1 1 TRK
SYRC0002 RDACRJ.DEGA.ALTTB.SR000001.P002 SYSDA 1 1 1 1 TRK
SYRC0003 RDACRJ.DEGA.ALTTB.SR000001.P003 SYSDA 1 1 1 1 TRK
SYRC0004 RDACRJ.DEGA.ALTTB.SR000001.P004 SYSDA 1 1 1 1 TRK
SYSDA 1 1 1 1 TRK
BMC50394I UNABLE TO LOCATE SORT WORK DATASETS, DDNAME = 'SORTWKNN'
BMC50474I BELOW 16M = 7492K, ABOVE 16M = 1405676K, CPUS = 3
BMC50479I TOTAL PAGES: 2402526, ALLOWED: 0; AVAILABLE PAGES: 257783, ALLOWED: 257783
BMC51302I MAX TASKS = 4, MAX PARTITIONS PER TASK = 1, SORTWKS PER TASK = 32, MAX OPEN
PARTITIONS PER TASK = 1
BMC50486I 1: BMCSORT STARTED, 256K BELOW 16M, 1280K TOTAL MEMORY, 0 PAGES HYPERSPACE
BMC50481I 1: SORT COMPLETE. ELAPSED TIME = 00:00:00
BMC50244I REGISTERING HEADER PAGE - 00000000 FOR DBID(4295) PSID(18)
BMC50245I REGISTERING HEADER PAGE - 00100000 FOR DBID(4295) PSID(18)
BMC50245I REGISTERING HEADER PAGE - 00200000 FOR DBID(4295) PSID(18)
BMC50245I REGISTERING HEADER PAGE - 00300000 FOR DBID(4295) PSID(18)
BMC50244I PROCESSING PAGESET DBID(4295) PSID(18) PART(1) THAT MIGHT CONTAIN VERSIONED DATA
BMC50486I 2: BMCSORT STARTED, 256K BELOW 16M, 1280K TOTAL MEMORY, 0 PAGES HYPERSPACE
BMC50244I REGISTERING HEADER PAGE - 00000000 FOR DBID(4295) PSID(18)
BMC50244I REGISTERING HEADER PAGE - 00100000 FOR DBID(4295) PSID(18)
BMC50244I REGISTERING HEADER PAGE - 00200000 FOR DBID(4295) PSID(18)
BMC50244I REGISTERING HEADER PAGE - 00300000 FOR DBID(4295) PSID(18)
BMC50244I PROCESSING PAGESET DBID(4295) PSID(18) PART(2) THAT MIGHT CONTAIN VERSIONED DATA
BMC50486I 3: BMCSORT STARTED, 256K BELOW 16M, 1280K TOTAL MEMORY, 0 PAGES HYPERSPACE
BMC50481I 3: SORT COMPLETE. ELAPSED TIME = 00:00:00
BMC50244I PROCESSING PAGESET DBID(4295) PSID(18) PART(4) THAT MIGHT CONTAIN VERSIONED DATA
BMC50486I 4: BMCSORT STARTED, 256K BELOW 16M, 1280K TOTAL MEMORY, 0 PAGES HYPERSPACE
BMC50244I REGISTERING HEADER PAGE - 00000000 FOR DBID(4295) PSID(18)
BMC50244I REGISTERING HEADER PAGE - 00100000 FOR DBID(4295) PSID(18)
BMC50244I REGISTERING HEADER PAGE - 00200000 FOR DBID(4295) PSID(18)
BMC50244I REGISTERING HEADER PAGE - 00300000 FOR DBID(4295) PSID(18)
BMC50244I PROCESSING PAGESET DBID(4295) PSID(18) PART(3) THAT MIGHT CONTAIN VERSIONED DATA
BMC50486I 4: BMCSORT STARTED, 256K BELOW 16M, 1280K TOTAL MEMORY, 0 PAGES HYPERSPACE
BMC50481I 4: SORT COMPLETE. ELAPSED TIME = 00:00:00
BMC50244I UNLOAD TASK COMPLETE. ELAPSED TIME = 00:00:00
**** B M C E X E C U T I O N   F O R   D B 2
****
C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  79
BMC50477I 0: PARTITION = 4, ROWS/KEYS = 1, I/O WAITS = 6 ,DDNAME = SYS00018
BMC50474I RELOAD STATISTICS: 1 ROWS/KEYS LOADED INTO PARTITION 4
BMC50476I DDNAME = SYDL0004, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I 6 PAGES COPIED TO DATASET = 'RDACRJ.ALTTB.T01AICP.P000.LP0001.P004'
BMC50476I DDNAME = SYRC0004, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50482I 0: BUILD COMPLETE. ELAPSED TIME = 00:00:00  DSN =
'DEGACAT.BMCDBD.CJBASIC.IRT01AICP.I0001.A002'
BMC50471I 0: PARTITION = 2, ROWS/KEYS = 0, I/O WAITS = 5 ,DDNAME = SYS00020
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEGACAT.BMCDBD.CJBASIC.IRT01AICP.I0001.A002'
BMC50482I 0: RELOAD COMPLETE. ELAPSED TIME = 00:00:00  DSN =
'DEGACAT.BMCDBD.CJBASIC.IRT01AICP.I0001.A002'
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEGACAT.BMCDBD.CJBASIC.IRT01AICP.I0001.A002'
BMC50482I 0: BUILD COMPLETE. ELAPSED TIME = 00:00:00  DSN =
'DEGACAT.BMCDBD.CJBASIC.IRT01AICP.I0001.A002'
BMC50477I 0: PARTITION = 3, ROWS/KEYS = 0, I/O WAITS = 5 ,DDNAME = SYS00022
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEGACAT.BMCDBD.CJBASIC.IRT01AICP.I0001.A003'
BMC50476I DDNAME = SYDL0003, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 5 PAGES COPIED TO DATASET = 'RDACRJ.ALTTB.T01AICP.P000.LP0001.P003'
BMC51276I BUILD  STATISTICS:  2 KEYS LOADED INTO INDEX 'CJBASIC.I_T01AICP1'
BMC51275I RELOAD STATISTICS:  2 ROWS LOADED INTO SPACE 'CJBASIC.T01AICP'
BMC50041I 0: ZIIP NOT ENABLED (0) USING XBM SUBSYSTEM XBMB
BMC50004I RELOAD PHASE COMPLETE. ELAPSED TIME = 00:00:05
BMC50891I 4: DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A001' IS RENAMED
BMC50891I 6: DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A002' IS RENAMED
BMC50891I 8: DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A004' IS RENAMED
BMC50890I 1: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A001'...
BMC50890I 4: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A004'...
BMC50890I 5: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A001'...
BMC50890I 6: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A002'...
BMC50890I 8: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A004'...
BMC50891I 2: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A002'...
BMC50891I 3: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A003'...
BMC50891I 4: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A004'...
BMC50890I 1: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A001'...
BMC50891I 2: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A002'...
BMC50891I 3: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A003'...
BMC50891I 4: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A004'...
BMC50890I 1: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A001'...
BMC50891I 2: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A002'...
BMC50891I 3: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A003'...
BMC50891I 4: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A004'...
BMC50890I 1: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A001'...
BMC50891I 2: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A002'...
BMC50891I 3: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A003'...
BMC50891I 4: ATTEMPTING TO RENAME DATASET 'DEGACAT.DSNDBC.CJBASIC.IRT01AICP.I0001.A004'...

Achieving your goals with ALTER and CHANGE MANAGER
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  81
BMC50890I 4: ATTEMPTING TO DELETE DATASET 'DEGACAT.OLDDBC.CJBASIC.T01AICP.I0001.A004'...
BMC50890I 1: ATTEMPTING TO DELETE DATASET 'DEGACAT.OLDDBC.CJBASIC.T01AICP.I0001.A001'...
BMC50890I 8: ATTEMPTING TO DELETE DATASET 'DEGACAT.OLDDBC.CJBASIC.IRT01AICP.I0001.A004'...
BMC50891I 2: DATASET 'DEGACAT.OLDDBC.CJBASIC.T01AICP.I0001.A002' IS DELETED
BMC50891I 5: DATASET 'DEGACAT.OLDDBC.CJBASIC.IRT01AICP.I0001.A001' IS DELETED
BMC50891I 7: DATASET 'DEGACAT.OLDDBC.CJBASIC.IRT01AICP.I0001.A003' IS DELETED
BMC50891I 4: DATASET 'DEGACAT.OLDDBC.CJBASIC.T01AICP.I0001.A004' IS DELETED
BMC50891I 1: DATASET 'DEGACAT.OLDDBC.CJBASIC.T01AICP.I0001.A001' IS DELETED
BMC50891I 6: DATASET 'DEGACAT.OLDDBC.CJBASIC.IRT01AICP.I0001.A002' IS DELETED
BMC50891I 3: DATASET 'DEGACAT.OLDDBC.CJBASIC.T01AICP.I0001.A003' IS DELETED

**** BMC EXECUTION FOR DB2

**** VERSION: V11.01.00

(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 12.55.30 ON SYSID DB2B WORKLIST EXECUTION LOG PAGE 14

------------------------------------------------------------------------------------------
(Line Prefix: Denotes Worklist Input)

STATEMENTS

BMC50891I 8: DATASET 'DEGACAT.OLDDBC.CJBASIC.IRT01AICP.I0001.A004' IS DELETED
BMC50501I DB2 OBJECT STATISTICS
BMC50515I TABLESPACE CJBASIC.T01AICP PARTS = 4 TABLES = 1 SEGSIZE = 0
PAGESIZE = 4K
BMC50516I PART NACTIVE CARD EXTENTS DBCARD PCOMP KSAVED PSAVED
BMC50517I 1 4 1 1 0 0 0 0
BMC50517I 2 3 0 1 0 0 0 0
BMC50517I 3 3 0 1 0 0 0 0
BMC50517I 4 4 1 1 0 0 0 0
BMC50518I TABLE CJBASIC.T_T01AICP
BMC50519I ROWAVG NPAGES CARD PCTPAGES
BMC50520I 210 6 2 42
BMC50521I INDEX CJBASIC.I_T01AICP1
BMC50526I FIRST KEY COLUMN = COLSI_2
BMC50527I CLUSTER = Y UNIQUE = D COMPRESS = N PAGESIZE = 4K KEYLEN = 19 COLCOUNT = 2
BMC50528I PART NACTIVE CARD EXTENTS LEVELS
BMC50529I 1 5 1 1 2
BMC50529I 2 5 0 1 2
BMC50529I 3 5 0 1 2
BMC50529I 4 5 1 1 2
BMC50529I DB2 REAL-TIME-STATISTICS -RESET STATS- FUNCTION FOR REORG UTILITY SUCCESSFUL FOR ALL OBJECTS
BMC50500I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
::SYNC 000400 REORG OF TABLESPACE CJBASIC.T01AICP COMPLETE 32382394
::SYNC 000450 END OF REORG TABLESPACE SECTION 02043851
:----------------------------------------
:: RNST 000500
:: RUNSTATS TABLESPACE CJBASIC.T01AICP
:: HISTORY NONE
:: INDEX (ALL)
:: KEYCARD
:: TABLE ALL
:: 72888461

BMC16815I PARAMETERS: DEGA,ALTIB
BMC16822I Control PASSED TO 'DSNUTILB'
BMC16898I DEBUG: CONNECTED WITH USER: RDACRJ
BMC16898I DEBUG: DSNUTILB 0

DSNU000I 015 12:55:30.56 DSNUGUTC - OUTPUT START FOR UTILITY, UTILID = ALTIB
DSNU1044I 015 12:55:30.65 DSNUGTIS - PROCESSING SYSSN AS EBCDIC

82 ALTER and CHANGE MANAGER for DB2 User Guide
DSNU050I 015 12:55:30.66 DSNUGUTC - RUNSTATS TABLESPACE CJBASIC.T01AICP HISTORY NONE INDEX(ALL) KEYCARD TABLE
**** BMC EXECUTION FOR DB2 INC.
**** VERSION: V11.01.00 (C) COPYRIGHT 1991 - 2013 BMC SOFTWARE.
2013-01-15 12.55.31 ON SYSID DB2B WORKLIST EXECUTION
LOG PAGE 15

STATEMENTS

DSNU610I  *DEGA 015 12:55:31.40 DSNUSUTP - SYSTABLEPART CATALOG UPDATE FOR CJBASIC.T01AICP SUCCESSFUL
DSNU610I  *DEGA 015 12:55:31.41 DSNUSUPT - SYSTABSTATS CATALOG UPDATE FOR CJBASIC.T_T01AICP SUCCESSFUL
DSNU610I  *DEGA 015 12:55:31.43 DSNUSUPC - SYSCOLSTATS CATALOG UPDATE FOR CJBASIC.T_T01AICP SUCCESSFUL
DSNU610I  *DEGA 015 12:55:31.43 DSNUSUTB - SYSTABLES CATALOG UPDATE FOR CJBASIC.T_T01AICP SUCCESSFUL
DSNU610I  *DEGA 015 12:55:31.44 DSNUSUCO - SYSCOLUMNS CATALOG UPDATE FOR CJBASIC.T_T01AICP SUCCESSFUL
DSNU610I  *DEGA 015 12:55:31.44 DSNUSUTS - SYSTABLESPACE CATALOG UPDATE FOR CJBASIC.T_T01AICP SUCCESSFUL
DSNU610I  *DEGA 015 12:55:31.52 DSNUSUIP - SYSINDEXPART CATALOG UPDATE FOR CJBASIC.I_T01AICP1 SUCCESSFUL
DSNU610I  *DEGA 015 12:55:31.52 DSNUSUPI - SYSINDEXSTATS CATALOG UPDATE FOR CJBASIC.I_T01AICP1 SUCCESSFUL
DSNU610I  *DEGA 015 12:55:31.53 DSNUSUPC - SYSCOLSTATS CATALOG UPDATE FOR CJBASIC.I_T01AICP1 SUCCESSFUL
DSNU610I  *DEGA 015 12:55:31.53 DSNUSUPD - SYSCOLDISTSTATS CATALOG UPDATE FOR CJBASIC.I_T01AICP1 SUCCESSFUL
DSNU610I  *DEGA 015 12:55:31.53 DSNUSUX - SYSINDEXES CATALOG UPDATE FOR CJBASIC.I_T01AICP1 SUCCESSFUL
DSNU610I  *DEGA 015 12:55:31.54 DSNUSUCD - SYSCOLDIST CATALOG UPDATE FOR CJBASIC.I_T01AICP1 SUCCESSFUL
DSNU010I 015 12:55:31.56 DSNUGBAC - UTILITY EXECUTION COMPLETE, HIGHEST RETURN CODE=0
:-SYNC 000550 STATS FOR TS CJBASIC.T01AICP COMPLETE 92514304
:-SYNC 000600 END OF TABLESPACE STATISTICS SECTION 25475249
:*******************************************************************************
::REBD 000650
:   REBIND TRIGGER PACKAGE (-
:   CJBASIC.TR01ICP1 +
:   ) -
:   PLANMGMT(OFF) -
:   EXPLAIN(NO) -
:   14516887
BMC16822I Control PASSED TO 'IKJEFT01'
BMC16898I DEBUG: CONNECTED WITH USER: RDACRJ
BMC16898I DEBUG: IKJEFT01 0

READY
DSN SYSTEM(DEGA)
DSN
   REBIND TRIGGER PACKAGE ( CJBASIC.TR01ICP1 )
PLANMGMT(OFF) EXPLAIN(NO)
DSNT254I *DEGA DSNTBRB2 REBIND OPTIONS FOR
DSN255I  *DEGA DSNTBRB2 REBIND OPTIONS FOR
PACKAGE = DEGA.CJBASIC.TR01ICP1().
SQLERROR      NOPACKAGE
CURRENTDATA   NO
DEGREE        1
DYNAMICRULES  BIND
NODEFER       PREPARE
REOPT         NONE
KEEPDYNAMIC   NO
IMMEDWRITE    NO
DBPROTOCOL    DRDA
OPTHINT
ENCODING      EBCDIC(00000)
PLANMGMT      OFF
PLANMGMTSCOPE STATIC
CONCURRENTACCESSRESOLUTION
EXTENDEDINDICATOR
PATH          "SYSIBM","SYSFUN","SYSPROC","RDACRJ"

DSN232I  *DEGA SUCCESSFUL REBIND FOR
PACKAGE = DEGA.CJBASIC.TR01ICP1().
DSN
END
:-REBD 000700
:    REBIND TRIGGER PACKAGE ( -
:      CJBASIC.TR01ICP2 +
:    ) -
:      PLANMGMT(OFF) -
:     EXPLAIN(NO) - 97675415
BMC16822I  Control PASSED TO 'IKJEFT01'
BMC16898I  DEBUG: CONNECTED WITH USER: RDACRJ
BMC16898I  DEBUG: IKJEFT01 0
READY
DSN  SYSTEM(DEGA)
DSN
PLANMGMT(OFF)  EXPLAIN(NO)
DSN254I  *DEGA DSNTBRB2 REBIND OPTIONS FOR

84  ALTER and CHANGE MANAGER for DB2 User Guide
Press END to return to the Create, Alter, or Drop DB2 Objects panel.

Press END to return to the Task List Menu panel.

Press END to return to the CHANGE MANAGER Main Menu.

This example shows that with CHANGE MANAGER you have the ability to quickly and accurately specify changes to your data structures, perform an in-
depth analysis of those changes to determine their effect on your subsystem, and execute SQL and utilities, while collecting statistics on your DB2 objects.

**Migrating DB2 objects**

Just as changing DB2 objects is a three-stage process, migrating DB2 objects from one environment to another within the same DB2 subsystem in ALTER or CHANGE MANAGER is also a three-stage process.

The three stages are specification, analysis, and execution.

1. **Specification**
   - Identify the objects to be migrated and specify changes to the data structures.

2. **Analysis**
   - Analyze the specified migration changes for accuracy and completeness. The result of the analysis stage is a worklist that contains the SQL and the information that is necessary to generate the data structure in the target DB2 system.

3. **Execution**
   - Execute the worklist that is generated during the analysis stage.

   **Note**
   The changes that you specify in the specification stage and analyze in the analysis stage are actually made to the objects in the execution stage.

The following example illustrates these three stages by showing how to use the Task List feature of CHANGE MANAGER to quickly and easily migrate objects from one environment to another environment within the same DB2 subsystem.

   **Note**
   Remember that CHANGE MANAGER can also migrate only the changes that you make to a data structure on one DB2 subsystem to other DB2 subsystems.

Before you begin a set of tasks to achieve a particular goal, you should allocate your JCL data sets for analysis and execution, ensure that you are using the correct DB2 subsystem, and define your job card with the correct account information.

For more information, view the Quick Course "CHANGE MANAGER for DB2 - Migrating in Same Subsystem."
To migrate objects

1 Start the Task List feature.

The Task List feature of CHANGE MANAGER enables you to achieve various goals by taking you step-by-step through a series of tasks.

a On the CHANGE MANAGER Main Menu (see Figure 5 on page 49), select Task List and press Enter.

The Task List Menu is displayed, as shown in Figure 25 on page 87.

Figure 25: Task List Menu

ALUFTASK ------------------------- Task List Menu -----------------------------
Command ===> 

Select a task. Then press Enter.
2 1. Create, alter, or drop DB2 objects
2. Migrate DB2 objects using Specification
3. Migrate DB2 objects using scope rules from a Migrate Profile

Commands: HELP END


The Migrate DB2 Objects Using Specification panel is displayed, as shown in Figure 26 on page 87.

Figure 26: Migrate DB2 Objects Using Specification panel

ALUFPALT -------------------- Migrate DB2 Objects Using Specification ---------
Command ===> 

Specify a Migrate WORKID.
If the WORKID does not exist, the Create WORKID panel will be displayed.
To see a list of WORKIDs, type a wildcard pattern. (For example BM*.*)

Migrate WORKID: RDACRJ.MIGDB

Optionally specify a Migrate Profile.
If the profile does not exist, the Create Profile panel will be displayed.
To see a list of profiles, type a wildcard pattern. For example BM*.*)

Migrate Profile:

Type S to select processing options. Then press Enter to continue.
  _ Create or edit a Migrate Profile
  _ Create or edit a WORKID
  _ Specify changes to DB2 objects
  _ Analyze the changes to create a worklist
  _ Execute the worklist to process the changes

Commands: HELP END
On the Migrate DB2 Objects Using Specification panel, specify a migrate-type work ID.

A work ID enables you to identify, control, and track a unit of work. The migrate-type work ID is used to specify data structure definitions to be migrated from one DB2 subsystem to another, or to duplicate structures within the same subsystem.

a. On the Migrate DB2 Objects Using Specification panel, type the name of a migrate-type work ID, and type S to select all of the processing options except Create or edit a Migrate Profile. Then, press Enter.

The Create WORKID panel is displayed, as shown in Figure 27 on page 88.

Figure 27: Create WORKID panel

b. On the Create WORKID panel, select Migrate for the Type. Type S to select the WORKID Change Rules. Then, press Enter.

The Change Rules panel is displayed, as shown in Figure 28 on page 88.

Figure 28: Change Rules panel

Change rules enable you to specify global changes to the name, owner, creator, or other attributes of the objects. The new attributes that you specify apply to all objects in the migration that possess the attributes.

c. On the Change Rules panel, type E in the Act field to select Edit. Press Enter.
The Edit Change Rules panel is displayed, as shown in Figure 29 on page 89.

**Figure 29: Edit Change Rules panel**

![Edit Change Rules panel](image)

On the Edit Change Rules panel, select **Change** as the **Option** for the attribute, select **All** as the **Object**, and type **S** to select **Display Attribute List**. Press Enter.

By selecting All objects, you can specify one change rule that changes an attribute for all of the tables, indexes, views, and so on. You can change an attribute that applies to all of the migrated objects that have that attribute.

The Edit Rules Attribute Selection panel is displayed, as shown in Figure 30 on page 89.

**Figure 30: Edit Rules Attribute Selection panel**

![Edit Rules Attribute Selection panel](image)
e On the Edit Rules Attribute Selection panel, select the number for the Name attribute. Press Enter, and then press END.

The Edit Change Rules panel is displayed, as shown in Figure 31 on page 90.

**Figure 31: Edit Change Rules panel—NAME attribute**

```
ALUFORLE ---------------------- Edit Change Rules -----------------------------
Command ===> 
                4. Table             10. Trigg er            16. Sequences
                5. Check Constraint 11. Alias              17. All
                6. Foreign Key       12. Synonym
Attribute. . NAME                   _  Display Attribute List
Current Attribute Value. .
New Attribute Value. . .
Specify optional object name(s).
Part1                                                  (SG, DB, owner, schema)
Part2                                                  (TS, name)
Part3                                                  (SP, version)
Commands:  HELP END CANCEL PF4=ZOOM
```

f On the Edit Change Rules panel, type the Current Attribute Value and the New Attribute Value for the database name. Press END.

The change rules are displayed in the Change Rules panel, as shown in Figure 32 on page 90.

**Figure 32: Change Rules panel—NAME attribute**

```
ALUFRULL ------------------ Change Rules ------------------------------------
Command ===> SCROLL ===> PAGE
WORKID . . . : RDACRJ.MIGDB
Type action next to existing rule or overtype existing values.
E=Edit I=Insert D=Delete L=Like C=Create Rules 1 to 1 of 1
Act Opt Obj Attribute  Current Attribute Value     New Attribute Value
******************************************************************************
C   **  NAME       R1BASIC                     RJBASIC
******************************************************************************
```

On the Change Rules panel, type L in the Act field for the name rule. The L enables you to create a new rule like the existing rule. Press Enter.

h On the second line, type OWNER over NAME in the Attribute field, type * for the Current Attribute Value (if you do not know the current owner), and type your TSO logon ID (or other desired value) for the New Attribute Value. Press Enter.
The Change Rules panel is displayed, as shown in Figure 33 on page 91.

**Figure 33: Change Rules panel—OWNER attribute**

| ALURFRULL ------------------ | Change Rules ----------------------------------- |
| Command ====>              | SCROLL ===> PAGE                               |
| WORKID . . . : RDACRJ.MIGDB | Rules 1 to 2 of 2                              |
| Type action next to existing rule or overtype existing values. | More: >                                      |
| E=Edit I=Insert D=Delete L=Like C=Create | Current Attribute Value New Attribute Value |
| Act Opt Obj Attribute      | ************************************** TOP ************************************ |
| C ** NAME                  | C ** OWNER RIBASIC RJBASIC                    |
| Current Attribute Value    | ************************************* BOTTOM ********************************** |

i Press END.

The Edit WORKID Migration Options panel is displayed.

j Press END to return to the Object Specification panel, as shown in Figure 34 on page 91.

**Figure 34: Object Specification panel**

| ALUSOS GRATE ---------------------- | Object Specification ---------------------- |
| Command ====>                      | Changed Objects List                        |
| WORKID . . . : RDACRJ.MIGDB         | Specify the object type(s) to be included in a list. |
| Stogroup. .                        | (SG Name)                                   |
| Database. .                        | (DB Name)                                   |
| Tablespace                         | (DB.TS Name)                                |
| Table . .                          | (Own.TB Name)                               |
| Check Cnst.                        | (Own.TB.Chk )                               |
| Foreign Key                        | (Own.TB.Rel )                               |
| Index . .                          | (Own.IX Name)                               |
| Unique Cnst                        | (Own.TB.Uniq)                               |
| View . .                           | (Own.VW Name)                               |
| Trigger . .                        | (Sch.TR Name)                                |
| Alias . .                          | (Own.AL Name)                               |
| Synonym . .                        | (Own.SY Name)                               |

The following objects are retrieved independently of all other DB2 objects.

Sequence .                        (Sch.SQ Name)
StoredProcedure .                  (Sch.SP Name)

3 Identify the objects and dependent objects that you want to migrate.

a On the Object Specification panel, type the name of a Database. Press Enter.
The Mixed List panel is displayed, as shown in Figure 35 on page 92.

**Figure 35: Mixed List panel**

On the Mixed List panel, type **Z** in the Act field next to the database. Press Enter.

The Mixed List panel is displayed with the dependent objects of the database, as shown in Figure 36 on page 92.

**Figure 36: Mixed List panel with dependent objects**

On the Mixed List panel, type **M** in the Act field next to the database that you want to migrate. Press Enter.

Type **MO** in the Act field next to the database that you selected. Press Enter.
The Migrate Options Overrides panel is displayed, as shown in Figure 37 on page 93.

Figure 37: Edit WORKID Migration Options panel

On the Migrate Options Overrides panel, type **Y** to select the dependent objects that you want to migrate. Press **Enter**, and then press **END**.

In this example, data was selected. No dependent objects were selected.

The Mixed List panel is displayed, as shown in Figure 38 on page 93. An "*M" is displayed next to the database that you selected to migrate.

Figure 38: Mixed List panel—database selected for migration

On the Mixed List panel, type **M** in the **Act** field next to the table space that you want to migrate. Press **Enter**.
Type **MO** in the **Act** field next to the table space that you selected. Press **Enter**.

The Migrate Options Overrides panel is displayed.

On the Migrate Options Overrides panel, type **Y** to select the dependent objects that you want to migrate. Press **Enter**, and then press **END**.

In this example, data and all of the dependent objects were selected.

The Mixed List panel is displayed, as shown in Figure 39 on page 94. An "**M**" is displayed next to the table space that you selected to migrate.

**Figure 39: Mixed List panel—objects selected for migration**

```
ALUSMXL RATE --------------------------- Mixed List ---------------------------
Command ===> Scroll . . PAGE
WORKID . . . : RDACRJ.MIGDB
Commands: CANCEL
Type action next to object and press Enter.
E=Edit  L=Like  D=Drop  U=Undo  M=Migrate  MO=Migrate options
Objects 1 to 13 of 52
More:     +
Act Object-Type Objects
************************************ TOP **************************************
*M  DB . . . . . R1BASIC TST10AIM
    TB . . . . . R1BASIC T_T10AIM
    IX . . . . . RIBASIC I_T10AIM1
*M  TSI . . . . . R1BASIC T01AICP
    TB . . . . . R1BASIC T_T01AICP
    IXC . . . . . RIBASIC I_T01AICP1
    #TR . . . . . RIBASIC T01ICP1
    #TR . . . . . RIBASIC T01ICP2
    TS . . . . . R1BASIC T02ASSEG
    TB . . . . . R1BASIC T02ASSEG
    IX . . . . . RIBASIC I_T02ASSEG_P
    IX . . . . . RIBASIC I_T02ASSEG_X
```

Press **END** to return to the Object Specification panel. Press **END** again.
The Analysis Migrate Worklist Interface panel is displayed, as shown in Figure 40 on page 95.

**Figure 40: Analysis Migrate Worklist Interface panel**

ACMFANL2 --------------- Analysis Migrate Worklist Interface ------------------
Command ===> WORKID . . . . . : RDACRJ.MIGDB
Analyze "Migrate" WORKID

Select type of analysis.
1 1. Analyze Specification entries (use the WORKID's change rules)
   2. Analyze Specification entries (use a Migrate Profile's locations and change rules)
   3. Analyze Migrate Profile scope (use the Migrate Profile's locations and change rules)

Type an Outbound Migrate Profile, or type a wildcard pattern for a selection list if analysis type is 2 or 3.
Migrate Profile . .

Select run type. Then press Enter.
   Run Type . . . . 2 1. Foreground
   2. Batch
Commands: HELP END

4 Analyze the migration changes that you specified.

The Analysis component of the product examines the changes that you specified and generates a worklist that contains the instructions that are necessary to implement the actions chosen during the specification stage.

a On the Analysis Migrate Worklist Interface panel, select the **Analyze Specification entries (use the WORKID’s change rules)** option. Then select to run the Analysis in **Batch**. Press Enter.

*Note*
If you run Analysis in foreground, your TSO session is unavailable until the processing has completed.
The Analysis JCL Processing Interface panel is displayed, as shown in Figure 41 on page 96.

Figure 41: Analysis JCL Processing Interface panel

ACMFANL3 ---------------- Analysis JCL Processing Interface -------------------
Command ===>
WORKID . . . . : RDACRJ.MIGDB

Specify Dataset Names
JCL . . . . . . . . 'ACM.DB2V10.V102.ANLYJCL(MIGDB)'
Worklist . . . . . . . . 'ACM.DB2V10.V102.WLABASE01(MIGDB)'
Diagnostics . . . . SYSOUT

Select batch processing options. Then press Enter.
S Override
S Create JCL
S Edit JCL
- Submit JCL
- Edit Worklist
Commands: BROWSE PREVIOUS HELP END

b On the Analysis JCL Processing Interface panel, specify your data set names, and type S to select the processing options. In this example, one of the options for analysis is overridden. Press Enter.

Tip
To change the default values, see the section explaining overrides in the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2

The Analysis Worklist Options panel is displayed, as shown in Figure 42 on page 96.

Figure 42: Analysis Worklist Options panel

-------------------------- Analysis Worklist Options --------------------------
Command ===>
Type information. Then press Enter to continue.

Specify allocation parameters for new datasets.
Dataset    Unitname   Volume   Priqty   Secqty   Alloc Unit
Worklist   SYSDA                   15        5   Tracks

Specify optional Global AUTHID.
Global AUTHID ..

Specify sync point frequency.
Frequency . . . . 10

Analyze for ROLE as OWNER.
ROLEOWN    .  .  . N

Commands: HELP END

c On the Analysis Worklist Options panel, press Enter to accept the defaults.
Tip
To change the default values, see the section explaining overrides in the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.

The Analysis Worklist Command Options panel is displayed, as shown in Figure 43 on page 97.

Figure 43: Analysis Worklist Command Options panel

On the Analysis Worklist Command Options panel, press Enter to accept the defaults.

The Analysis Input Stream Parameters panel is displayed, as shown in Figure 44 on page 97.

Figure 44: Analysis Input Stream Parameters panel

On the Analysis Input Stream Parameters panel, press Enter to accept the defaults.
On the Analysis Input Stream Parameters panel, type S to select the option **Omit the -STOP command between phases of migrate worklist.** When you migrate within a single subsystem, you normally do not want to stop between phases in a migrate worklist. (For more information, see the *ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.*)

Press END.

The JCL Generation component is invoked and Batch Analysis JCL Generation is displayed, as shown in Figure 45 on page 98. The JCL includes the //ALUIN DD and //AJXPOFIN DD statements (input streams), which include keywords for processing the worklist. For more information about the input streams, see the ALTER and CHANGE MANAGER for DB2 Reference Manual.

---

**Figure 45: Batch Analysis JCL**

```jcl
//RDACRJA JOB (5715), 'ANALYSIS-MIGDB',
//  CLASS=A, MSGCLASS=X, MSGLEVEL=(1,1),
//  NOTIFY=RDACRJ
//*ROUTE XEQ BMCPLEX1
//JOBPARM SYSAFF=DB2B
//*****************************************************************************
// ** CREATED BY : RDACRJ
// ** TIMESTAMP : 01/15/2013.14.24.37
// ** ENVIRONMENT: ISPF 6.3 MVS TSO
// ** RELEASE : V11.01.00 06/25/2013
// ** DB2 VERSION: 10015
//*****************************************************************************
// **-------------------------------------------------------------------
// **         BMC CHANGE MANAGER ANALYSIS
// **-------------------------------------------------------------------
//ANALYSIS EXEC PGM=ALUAMAIN,
//  REGION=0M,
//  PARM='JMB1SEGA'
//STEPLIB DD DSN=SYS3.DEGA.DSNEXIT,DISP=SHR
//   DD DSN=CSGI.DB2V10.SDSNL0AD,DISP=SHR
//*
//ABNLIGNR DD DUMMY
//SYSUDUMP DD SYSOUT=* //ALUPRINT DD SYSOUT=* //AJXPOFVL DD DUMMY
//ALUIN DD *
SSID DEGA
WORKID RDACRJ.MIGDB
INCLUDE (DATA SQL AMS )
SORTDEV SYSDA
SMSINCLUDE ( )
BMCREBUILD
STANDALONESTATS DB2STATSUPD TABLEALL
NOHISTORY
BMCCOPY
BMCLOAD NOTULCOPY
BMCFASTLOAD
MAXSYSUT 20
DYNUNLD
DYNCOPY
NOPARTCOPY
COPYDDN (COPY01 )
REGISTER (COPY01 )
NOREGENIDENTITY
SINGLEPHASE
```
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  99
Achieving your goals with ALTER and CHANGE MANAGER

DEF_GDG_NOSCR = N
DIAG_MSGCLASS=X
DISC_DATACLASS_ALT=
DISC_DATACLASS=
DISC_EXPDT=
DISC_MGMTCLASS_ALT=
DISC_MGMTCLASS=
DISC_PREFIX=&PREFIX..&WKID
DISC_PRIQTY=10
DISC_RETPD=
DISC_SECQTY=2
DISC_STORCLASS_ALT=
DISC_STORCLASS=
DISC_THRESH=0
DISC_UNIT_ALT=
DISC_UNIT=SYSDA
GDG_MODEL=SYS1.MODEL
PCPY1_DATACLASS_ALT=
PCPY1_DATACLASS=
PCPY1_EXPDT=
PCPY1_MGMTCLASS_ALT=
PCPY1_MGMTCLASS=
PCPY1_PREFIX=&PREFIX..&WKID..&SPNAME..P&PART
PCPY1_PRIQTY=10
PCPY1_RETPD=
PCPY1_SECQTY=2
PCPY1_STACK=N
PCPY1_STORCLASS=
PCPY1_STORCLASS_ALT=
PCPY1_SUPPRESS_SUFF=N
PCPY1_THRESH=0
PCPY1_UNIT_ALT=
PCPY1_UNIT=SYSDA
PCPY2_DATACLASS_ALT=
PCPY2_DATACLASS=
PCPY2_EXPDT=
PCPY2_MGMTCLASS_ALT=
PCPY2_MGMTCLASS=
PCPY2_PREFIX=&PREFIX..&WKID..&SPNAME..P&PART
PCPY2_PRIQTY=10
PCPY2_RETPD=
PCPY2_SECQTY=2
PCPY2_STACK=N
PCPY2_STORCLASS=
PCPY2_STORCLASS_ALT=
PCPY2_SUPPRESS_SUFF=N
PCPY2_THRESH=0
PCPY2_UNIT_ALT=
PCPY2_UNIT=SYSDA
PUNCH_DATACLASS=
PUNCH_EXPDT=
PUNCH_MGMTCLASS=
PUNCH_MGMTCLASS_ALT=
PUNCH_SUPPRESS_SUFF=N
PUNCH_THRESH=0
PUNCH_UNIT_ALT=
PUNCH_UNIT=SYSDA
RCPY1_DATACLASS_ALT=
RCPY1_DATACLASS=
RCPY1_EXPDT=
RCPY1_MGMTCLASS_ALT=
RCPY1_MGMTCLASS=
RCPY1_PREFIX=&PREFIX..&WKID..&SPNAME..P&PART
RCPY1_PRIQTY=10
RCPY1_RETPD=
RCPY1_SECQTY=2
RCPY1_STACK=N
RCPY1_STORCLASS=
RCPY1_STORCLASS_ALT=
RCPY1_SUPPRESS_SUFF=N
RCPY1_THRESH=0
RCPY1_UNIT_ALT=
RCPY1_UNIT=SYSDA
RCPY2_DATACLASS_ALT=
RCPY2_DATACLASS=
RCPY2_EXPDT=
RCPY2_MGMTCLASS_ALT=
RCPY2_MGMTCLASS=
RCPY2_PREFIX=&PREFIX..&WKID..&SPNAME..P&PART
RCPY2_PRIQTY=10
RCPY2_RETPD=
RCPY2_SECQTY=2
RCPY2_STACK=N
RCPY2_STORCLASS=
RCPY2_STORCLASS_ALT=
RCPY2_SUPPRESS_SUFF=N
RCPY2_THRESH=0
RCPY2_UNIT_ALT=
RCPY2_UNIT=SYSDA
REORG_MAPTAB=JEFF.JFLTBMAP
SRTOUT_DATACLASS_ALT=
SRTOUT_DATACLASS=
SRTOUT_EXPDT=
SRTOUT_MGMTCLASS_ALT=
SRTOUT_MGMTCLASS=
SRTOUT_PREFIX=&PREFIX..&WKID..&STEPN
SRTOUT_PRIQTY=10
SRTOUT_RETPD=
SRTOUT_SECQTY=2
SRTOUT_STORCLASS_ALT=
SRTOUT_STORCLASS=
SRTOUT_UNIT_ALT=
SRTOUT_UNIT=SYSDA
SYSUT_DATACLASS_ALT=
SYSUT_DATACLASS=
SYSUT_EXPDT=
SYSUT_MGMTCLASS_ALT=
SYSUT_MGMTCLASS=
SYSUT_PREFIX=&PREFIX..&WKID..&STEPN
SYSUT_PRIQTY=10
SYSUT_RETPD=
SYSUT_SECQTY=2
SYSUT_STORCLASS_ALT=
SYSUT_STORCLASS=
SYSUT_THRESHOLD=0
SYSUT_UNIT_ALT=
SYSUT_UNIT=SYSDA
TAPE_EXPDT=
TAPE_RETPD=
TAPE_VOLCNT=99
TAPE1=CARTVTS
TAPE2=CARTVTS
TAPE3=CARTVTS
TEMPUNIT=SYSDA
UNLD1_DATACLASS_ALT=
UNLD1_DATACLASS=
UNLD1_EXPDT=
UNLD1_MGMTCLASS_ALT=
UNLD1_MGMTCLASS=
UNLD1_PREFIX=&PREFIX..&MSSID..&WORKID8
UNLD1_PRIQTY=10
UNLD1_RETPD=
UNLD1_SECQTY=2
UNLD1_STACK=N
UNLD1_STORCLASS_ALT=
UNLD1_STORCLASS=
UNLD1_SUPPRESS_SUFF=N
UNLD1_THRESH=0
UNLD1_UNIT_ALT=
UNLD1_UNIT=SYSDA
UNLD2_DATACLASS_ALT=
UNLD2_DATACLASS=
UNLD2_EXPDT=
UNLD2_MGMTCLASS_ALT=
UNLD2_MGMTCLASS=
UNLD2_PREFIX=&PREFIX..&MSSID..&WORKID8
UNLD2_PRIQTY =10
UNLD2_RETPD=
UNLD2_SECQTY =2
UNLD2_STACK=N
UNLD2_STORCLASS_ALT=
UNLD2_STORCLASS=
UNLD2_SUPPRESS_SUFF=N
UNLD2_THRESH=0
UNLD2_UNIT_ALT=
UNLD2_UNIT=SYSDA
UNLD3_DATACLASS=
UNLD3_EXPDT=
UNLD3_MGMTCLASS_ALT=
UNLD3_MGMTCLASS=
UNLD3_PREFIX=&PREFIX..&MSSID..&WORKID8
UNLD3_SUPPRESS_SUFF=N
UNLD3_UNIT=SYSDA
UNLD4_PREFIX=&PREFIX..&MSSID..&WORKID8
UNLD4_STORCLASS=
UNLD4_UNIT=SYSDA
UNLD4_MGMTCLASS=
UNLD4_DATACLASS=
UNLD4_SUPPRESS_SUFF=N
UNLD4_FREF_DATACLASS=
UNLD4_FREF_MGMTCLASS=
UNLD4_FREF_PREFIX=&PREFIX..&MSSID..&WORKID8
UNLD4_FREF_STORCLASS=
UNLD4_FREF_UNIT=SYSDA
UNLD4_FREF_PRIQTY=10
UNLD4_FREF_DIRBLOCK=250
UNLD4_FREF_SECQTY=2
UNLD4_FREF_SUPPR_SUFF=N
USER_VAR1_CHAR=
USER_VAR2_CHAR=
USER_VAR3_CHAR=
USER_VAR4_CHAR=0
USER_VAR5_CHAR=0

```sql
//WORKL001 DD DISP=SHR,
//   DSN=ACM.DB2V10.V102.WLBASE01(MIGDB)
//*--------------------------------------------------------------------
//* END OF JOB
//*--------------------------------------------------------------------
```

102  ALlTER and CHANGE MANAGER for DB2 User Guide
g Press END to return to the Analysis JCL Processing Interface panel, as shown in Figure 46 on page 103.

**Figure 46: Analysis JCL Processing Interface panel**

```
ACMFANL3 ---------------- Analysis JCL Processing Interface -------------------
Command   ====>
WORKID . . . . : RDACRJ.MIGDB

Specify Dataset Names
   JCL . . . . . . . . 'ACM.DB2V10.V102.ANLYJCL(MIGDB)'
   Worklist . . . . . 'ACM.DB2V10.V102.WLBASE01(MIGDB)'
   Diagnostics . . . . SYSOUT

Select batch processing options. Then press Enter.
* Override
* Create JCL
* Edit JCL
S Submit JCL
  Edit Worklist
Commands:  BROWSE PREVIOUS HELP END
```

h On the Analysis JCL Processing Interface panel, press Enter to submit the JCL.

i After you submit the job, press Enter to return to the Analysis JCL Processing Interface panel.

The results of Analysis are a diagnostic output file (ALUPRINT) and a worklist.

j To review the results of the analysis of the work ID (ALUPRINT), you can split your ISPF session or you can use your normal method to review SYSOUT. The
last message in the diagnostic output (Figure 47 on page 104) indicates that the worklist completed successfully.

**Figure 47: Analysis diagnostic output**

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSID DEGA</td>
</tr>
<tr>
<td>WORKID RDACRJ.MIGDB</td>
</tr>
<tr>
<td>INCLUDE (DATA SQL AMS )</td>
</tr>
<tr>
<td>SORTDEV SYSDA</td>
</tr>
<tr>
<td>SMSINCLUDE ( )</td>
</tr>
<tr>
<td>BMCREBUILD</td>
</tr>
<tr>
<td>STANDBASESTATS DB2STATSUPD TABLEALL</td>
</tr>
<tr>
<td>NOHISTORY</td>
</tr>
<tr>
<td>BMCCOPY</td>
</tr>
<tr>
<td>BMLOAD NOUTILCOPY</td>
</tr>
<tr>
<td>BMCFLOAD</td>
</tr>
<tr>
<td>MAXSYSUT 20</td>
</tr>
<tr>
<td>DYNUNLOD</td>
</tr>
<tr>
<td>DYNCOPY</td>
</tr>
<tr>
<td>NOPARTCOPY</td>
</tr>
<tr>
<td>COPYDDN (COPY01 )</td>
</tr>
<tr>
<td>REGISTER (COPY01 )</td>
</tr>
<tr>
<td>NOREGENIDENTITY</td>
</tr>
<tr>
<td>SINGLPHASE</td>
</tr>
<tr>
<td>NOVALPROP</td>
</tr>
<tr>
<td>BMCCUNLOAD NOUNLDCOLL</td>
</tr>
<tr>
<td>BMCREORG IBMCHECK</td>
</tr>
<tr>
<td>SYMPHPOIN 10</td>
</tr>
<tr>
<td>NODBRMLIB</td>
</tr>
<tr>
<td>ORDERBY</td>
</tr>
<tr>
<td>UNLOADTABLE</td>
</tr>
<tr>
<td>UNLOADEMPTY</td>
</tr>
<tr>
<td>NOSTOPCOMMIT</td>
</tr>
<tr>
<td>TABLEACCESS</td>
</tr>
<tr>
<td>NOPARALLEL</td>
</tr>
<tr>
<td>NOERGONLINE</td>
</tr>
<tr>
<td>KEYCARD</td>
</tr>
<tr>
<td>REORGREF</td>
</tr>
<tr>
<td>DYNEORPOL</td>
</tr>
<tr>
<td>CLONEDATA</td>
</tr>
<tr>
<td>*BMC396243I DB2 VERSION 10 NFM</td>
</tr>
<tr>
<td>*** PTS APPPLIED TO PRODUCT:***</td>
</tr>
<tr>
<td>*** WORKLIST GENERATION 2013-01-15-14.32.21.382780</td>
</tr>
<tr>
<td>*** CHANGE MANAGER FOR DB2 VERSION 11.01.00(06/25/2013)</td>
</tr>
<tr>
<td>WORKLIST GENERATION EXECUTION FOR WORKID RDACRJ.MIGDB ON SSID DEGA</td>
</tr>
<tr>
<td>LIMIT KEY VALIDATION STARTING, COMPILED Dec 20 2012 AT 21:39:04</td>
</tr>
<tr>
<td>LIMIT KEY VALIDATION COMPLETE</td>
</tr>
<tr>
<td>*BMC396243I DB2 VERSION 10 NFM</td>
</tr>
<tr>
<td>*** THE FOLLOWING MESSAGES ARE WARNINGS ONLY ***</td>
</tr>
<tr>
<td>*BMC45339W TAPES CANNOT BE STACKED AND MULTIPLE TAPE DRIVES MAY BE REQUIRED IF MULTITASKING IS USED TO UNLOAD AND LOAD</td>
</tr>
<tr>
<td>* PARTITIONED TABLE SPACES. TO DISABLE MULTITASKING AND TO ENABLE TAPE STACKING, ADD THE NOMTASK KEYWORD TO THE ALUIN INPUT STREAM.</td>
</tr>
<tr>
<td>*** END OF WARNING MESSAGES - WARNINGS WERE ISSUED ***</td>
</tr>
<tr>
<td>WORKLIST WAS WRITTEN TO DATASET 'ACM.DB2V10.V102.WLBASE01(MIGDB)'</td>
</tr>
<tr>
<td>WORKLIST GENERATION COMPLETE WORKID RDACRJ.MIGDB</td>
</tr>
</tbody>
</table>

**Figure 48 on page 104** shows the worklist that was written to the data set that you specified on the Analysis JCL Processing Interface panel.

**Figure 48: CHANGE MANAGER migrate-type worklist**

*** CHANGE MANAGER FOR DB2 VERSION 11.01.00(06/25/2013)  

---

104  *ALTER and CHANGE MANAGER for DB2 User Guide*
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  105

*** PTFS APPLIED TO PRODUCT:
*** NONE
***********************************************************************
* ALUIN PARMS:
*   SSID DEGA
*   WORKID RDACRJ.MIGDB
*   INCLUDE (DATA SQL AMS )
*   SORTDEV SYSDA
*   SMSINCLUDE ( )
*   BMCREBUILD
*   STANDALONESTATS DB2STATSUPD TABLEALL
*   NOHISTORY
*   BMCCOPY
*   BMLOAD  NOUTILCOPY
*   BMCFASTLOAD
*   MAXSYSUT 20
*   DYNUNLD
*   DYNCOPY
*   NOPARTCOPY
*   COPYDDN (COPY01 )
*   REGISTER (COPY01 )
*   NOREGENIDENTITY
*   SINGLEPHASE
*   NOVALPROP
*   BMUNLOAD NOUNLDCOLL
*   BMCREORG IBMCHECK
*   SYNCPOINT 10
*   NODBRMLIB
*   ORDERBY
*   UNLOADTABLE
*   UNLOADEMPTY
*   NOSTOPCOMMIT
*   TABLEACCESS
*   NOPARALLEL
*   NOREORGONLINE
*   KEYCARD
*   REORGALL
*   REORGREF
*   DYNREORG
*   CLONEDATA
***********************************************************************
- TIME 000050 '2013-01-15-14.32.21.382780'
- SSID 000100 DEGA
- JCLP 000150 MIGR MSSID DEGA
- WKID 000200 RDACRJ.MIGDB
***********************************************************************
***********************************************************************
*** THE FOLLOWING MESSAGES ARE WARNINGS ONLY ***
***********************************************************************
*BMC45339W TAPES CANNOT BE STACKED AND MULTIPLE TAPE DRIVES MAY BE
* REQUIRED IF MULTITASKING IS USED TO UNLOAD AND LOAD
* PARTITIONED TABLE SPACES. TO DISABLE MULTITASKING AND TO
* ENABLE TAPE STACKING, ADD THE NOMTASK KEYWORD TO THE ALUIN
* INPUT STREAM.
***********************************************************************
*** END OF WARNING MESSAGES - WARNINGS WERE ISSUED ***
***********************************************************************
*** ORIGINATOR OF WORKLIST WAS WORKID RDACRJ.MIGDB ON SSID DEGA ***
- MIGR 000250 PHASE-1  START OF SECTION TO BE RUN ON SENDING SYSTEM
- SYNC 000300 START OF EXECUTION SYNCPOINT
***********************************************************************
-JCLP 000350 BMCD IPPARTS 4
-BMCD 000400
UNLOAD UNLOADDDN (SYSR)
ACTIVE (YES,NO)
ON FAILURE ALL TERMINATE UTILITY
DELETEFILES YES
DISCARDS 1
NULLTYPE T1 NULLCHAR ?
FORMAT BMCLOAD

SELECT *
FROM RIBASIC.T_T01AICP
ORDER BY
COLSI_2,
COLVC_8
;

OUTPUT SYSR
UNIT SYSDA
DSNAME 'RDACRJ.DEGA.MIGDB.SR000001.P&PART'

-SYNC 000450 UNLOAD TB RIBASIC.T_T01AICP COMPLETE
-SYNC 000500 END OF UNLOAD TABLES SECTION
******************************************************************************
-SETS 000550 SET CURRENT SQLID = 'RDACRJ'
-SQL 000600 CREATE DATABASE RJBASIC
   STOGROUP SYSDEFLT
   BUFFERPOOL BP8K0

-SETS 000650 SET CURRENT SQLID = USER
-SYNC 000700 END OF DATABASE SECTION
******************************************************************************
-SETS 000750 SET CURRENT SQLID = 'RDACRJ'
-SQL 000800 CREATE TABLESPACE T01AICP IN RJBASIC
   NUMPARTS 4
   (PART 1
    USING STOGROUP SYSDEFLT
    PRIQTY 440
    SECQTY 220
    PCTFREE 0
   , PART 2
    USING STOGROUP SYSDEFLT
    PRIQTY 440
    SECQTY 220
    PCTFREE 0
   , PART 3
    USING STOGROUP SYSDEFLT
    PRIQTY 440
    SECQTY 220
    PCTFREE 0
   , PART 4
    USING STOGROUP SYSDEFLT
    PRIQTY 440
    SECQTY 220
    PCTFREE 0
   )
   BUFFERPOOL BPO
   LOCKSIZE ANY
   LOCKMAX 10
   CLOSE NO
   SEGSIZE 0

-SETS 000850 SET CURRENT SQLID = USER
-SYNC 000900 END OF CREATE TABLESPACE SECTION
******************************************************************************
-SQL 000950 CREATE TABLE RJBASIC.T_T01AICP
   ( COLC_1
     CHAR(20) NOT NULL WITH DEFAULT
   , COLSI_2
     SMALLINT NOT NULL WITH DEFAULT
   , COLIN_3
     INTEGER NOT NULL WITH DEFAULT

106  ALTER and CHANGE MANAGER for DB2 User Guide
. COLSI_4
  SMALLINT NOT NULL WITH DEFAULT
. COLDC_5
  DECIMAL(6,2) NOT NULL WITH DEFAULT
. COLDC_6
  DECIMAL(9,0) NOT NULL WITH DEFAULT
. COLDC_7
  DECIMAL(5,2) NOT NULL WITH DEFAULT
. COLVC_8
  VARCHAR(15) NOT NULL WITH DEFAULT
. COLF1_9
  REAL NOT NULL WITH DEFAULT
. COLF2_10
  FLOAT NOT NULL WITH DEFAULT
. COLVC_11
  VARCHAR(15) NOT NULL WITH DEFAULT
. COLDT_12
  DATE NOT NULL WITH DEFAULT
. COLTS_13
  TIMESTAMP NOT NULL WITH DEFAULT
. COLC_14
  CHAR(100) NOT NULL WITH DEFAULT
)

CCSID EBCDIC
IN RJBASIC.T01AICP

-SQL 001000 COMMENT ON TABLE RJBASIC.T_T01AICP
  IS 'T01- FACTS DO NOT CEASE TO EXIST BECAUSE THEY ARE IGNORED'

-SYNC 001050 END OF TABLE SECTION
**********************************************************************
-SQL 001100 CREATE INDEX
  RJBASIC.I_T01AICP1
  ON RJBASIC.T_T01AICP
  ( COLSI_2
    COLVC_8 )
  NOT PADDED
  CLUSTER
  ( PART 1 VALUES (10,'C23456789012')
    USING STOGROUP SYSDEFLT
    PRIQTY 1440
    SECOQT 720
    PCTFREE 0
  , PART 2 VALUES (13,'K23456789012')
    USING STOGROUP R1BASIC3
    PRIQTY 1440
    SECOQT 720
    PCTFREE 0
  , PART 3 VALUES (16,'R23456789012')
    USING STOGROUP R1BASIC4
    PRIQTY 1440
    SECOQT 720
    PCTFREE 0
  , PART 4 VALUES (MAXVALUE,MAXVALUE)
    USING STOGROUP SYSDEFLT
    PRIQTY 1440
    SECOQT 720
    PCTFREE 0
  )
  BUFFERPOOL BP1
  CLOSE YES

-SYNC 001150 END OF CREATE INDEX SECTION
**********************************************************************
-JCLP 001200 BMCL IPPARTS 4
```
-#BMCL 001250
LOAD DATA REPLACE
INDSN('RDACRJ.DEGA.MIGDB.SR000001.P001',
    'RDACRJ.DEGA.MIGDB.SR000001.P002',
    'RDACRJ.DEGA.MIGDB.SR000001.P003',
    'RDACRJ.DEGA.MIGDB.SR000001.P004')
    ERRDDN SYSER001
    DISCARDS 1
    DISCARDON SYSDS001
    DELETEFILES NO
    ORDER YES
    FORMAT BMCUNLOAD
    WORKDDN SYSUT001
    LOADDDN SORTP
    ENFORCE NO
    REDEFINE NO
    UNIQUEINTO YES
    INTO TABLE RJBASIC.T_T01AICP
    WHEN TABLE = 1

-#SYNC 001300 LOAD TB RJBASIC.T_T01AICP COMPLETE
-#SYNC 001350 END OF LOAD TABLES SECTION

-#BMCC 001400
OUTPUT DCPYLOCP
    UNIT SYSDA
    DSNAM &UID.MIGDB.&TS.P&PART.LP000001
    COPY TABLESPACE RJBASIC.T01AICP
    COPYDDN (DCPYLOCP)

-#SYNC 001450 COPY OF MULTIPLE TABLESPACES COMPLETE
-#SYNC 001500 END OF COPY SECTION

-#RNST 001550
RUNSTATS TABLESPACE RJBASIC.T01AICP
    HISTORY NONE
    INDEX (ALL)
    KEYCARD
    TABLE ALL

-#SYNC 001600 STATS FOR TS RJBASIC.T01AICP COMPLETE
-#SYNC 001650 END OF TABLESPACE STATISTICS SECTION

k  Press END.
```
The Execution JCL Build Interface panel is displayed, as shown in Figure 49 on page 109.

Figure 49: Execution JCL Build Interface panel

The Execution component executes the migrate worklist in a single phase. First, the data is unloaded from the objects. Then, the objects are created, the data is loaded, and the utilities are executed.

The Execution component of the product uses the worklist as its input job stream and performs the tasks contained in the worklist. The worklist acts as a detailed procedure for implementing the changes that you defined in the specification stage.

a On the Execution JCL Build Interface panel, press **Enter** to accept the defaults.
The Execution Pre- and Post-Processing Interface panel is displayed, as shown in Figure 50 on page 110.

**Figure 50: Execution Pre- and Post-Processing Interface panel**

<table>
<thead>
<tr>
<th>ACMFEXC1</th>
<th>Execution Pre- and Post-Processing Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command ===&gt; WORKID . . . . . : RDACRJ.MIGDB</td>
<td></td>
</tr>
<tr>
<td>Specify additional steps to be included in this job. Then press Enter. Pre-Execution Compare</td>
<td></td>
</tr>
<tr>
<td>1. Do not do a compare before execution</td>
<td></td>
</tr>
<tr>
<td>2. Compare two previous baselines</td>
<td></td>
</tr>
<tr>
<td>3. Compare a previous baseline to the current catalog</td>
<td></td>
</tr>
<tr>
<td>Pre-Execution Baseline</td>
<td></td>
</tr>
<tr>
<td>1. Do not build a baseline before execution</td>
<td></td>
</tr>
<tr>
<td>2. Build a baseline before execution</td>
<td></td>
</tr>
<tr>
<td>Post-Execution Compare</td>
<td></td>
</tr>
<tr>
<td>1. Do not do a compare after execution</td>
<td></td>
</tr>
<tr>
<td>2. Compare a previous baseline to the new catalog structures</td>
<td></td>
</tr>
<tr>
<td>3. Compare the baseline built during a previous job step to the new catalog structures</td>
<td></td>
</tr>
<tr>
<td>4. Create CDL to fall back to a previous baseline</td>
<td></td>
</tr>
<tr>
<td>Post-Execution Baseline</td>
<td></td>
</tr>
<tr>
<td>1. Do not build a baseline after execution</td>
<td></td>
</tr>
<tr>
<td>2. Build a baseline after execution</td>
<td></td>
</tr>
<tr>
<td>Commands: HELP PREVIOUS END</td>
<td></td>
</tr>
</tbody>
</table>

b On the Execution Pre- and Post-Processing Interface panel, press Enter to accept the defaults.

The Execution JCL Processing Interface panel is displayed, as shown in Figure 51 on page 110.

**Figure 51: Execution JCL Processing Interface panel**

<table>
<thead>
<tr>
<th>ACMFEXC3</th>
<th>Execution JCL Processing Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command ===&gt; WORKID . . . . . : RDACRJ.MIGDB</td>
<td></td>
</tr>
<tr>
<td>Specify Dataset Names.</td>
<td></td>
</tr>
<tr>
<td>Execution JCL . . . . . 'ACM.DB2V10.V102.EXECJCL(MIGDB)'</td>
<td></td>
</tr>
<tr>
<td>Worklist . . . . . 'ACM.DB2V10.V102.WLBASE01(MIGDB)'</td>
<td></td>
</tr>
<tr>
<td>Diagnostics . . . . . SYSOUT</td>
<td></td>
</tr>
<tr>
<td>JCL Build Options.</td>
<td></td>
</tr>
<tr>
<td>Dataset Sizing : No Sizing (use Override Defaults to set or change)</td>
<td></td>
</tr>
<tr>
<td>Select Processing Options. Then press Enter to continue.</td>
<td></td>
</tr>
<tr>
<td>_ Override Defaults</td>
<td></td>
</tr>
<tr>
<td>_ Edit Worklist</td>
<td></td>
</tr>
<tr>
<td>S Build Execution JCL</td>
<td></td>
</tr>
<tr>
<td>S Edit Execution JCL</td>
<td></td>
</tr>
<tr>
<td>S Submit Execution JCL</td>
<td></td>
</tr>
<tr>
<td>Commands: BROWSE VIEWX PREVIOUS HELP END</td>
<td></td>
</tr>
</tbody>
</table>

c On the Execution JCL Processing Interface panel, specify your data set names, specify SYSOUT for diagnostics, accept the defaults for your build option, and type S to select your processing options. Press Enter.

The Execution JCL is displayed, as shown in Figure 52 on page 111. The JCL includes the //AEXIN DD statement (input stream), which includes keywords...
that are used in executing the JCL. For more information about the input stream, see the ALTER and CHANGE MANAGER for DB2 Reference Manual.

Figure 52: Execution JCL for a migrate-type worklist

```
//RDACRJE JOB (5715), 'EXECUTE-MIGDB',
// CLASS=A, MSGCLASS=X, MSGLEVEL=(1,1),
// NOTIFY=RDACRJ
/*ROUTE XEQ BMCPX1
/*JOBPARM SYSAFF=DB2B

//***************************************************
//*  CREATED BY :  RDACRJ
//*  TIMESTAMP  :  01/15/2013.15.00.19
//*  ENVIRONMENT:  ISPF 6.3MVS TSO
//*  RELEASE    :  V11.01.00 06/25/2013
//*  DB2 VERSION:  10015
//***************************************************

//*--------------------------------------------------------------------
//*        BMC CHANGE MANAGER EXECUTION
//*--------------------------------------------------------------------
//STEP1 EXEC PGM=AEXE MAIN,
//     REGION=0M, MEMLIMIT=NOLIMIT, PARM='JMB1SEGA',
//     COND=(4,LT)
//STEPLIB DD DSN=SYS3.DEGA.DSNEXIT, DISP=SHR
//        DD DSN=CSGI.DB2V10.SDSNLOAD, DISP=SHR
/*
//ABNLIGNR DD DUMMY
//SYSUDUMP DD SYSOUT=* 
//SYSTERM DD SYSOUT=* 
//ASUSRPRT DD SYSOUT=* 
//UTPRINT DD SYSOUT=* 
//*--------------------------------------------------------------------
//*  STATISTICS WORK DD STATEMENTS
//*--------------------------------------------------------------------
//STPRIN01 DD SYSOUT=* 
//RNPRIN01  DD SYSOUT=* 
//SYSOUT   DD SYSOUT=* 
//AJXERR   DD SYSOUT=* 
//AEXIN  DD *
ACM
SSID DEGA WORKID RDACRJ.MIGDB
DYNWORKUNIT SYSDA
DASDDOPT DSA2QDOP
CATDOPT DC102DOP
STOPWAIT 3
STOPWTSecs 10
LOCK EXCLUSIVE
//SYSIN  DD SPACE=(CYL,(15,15)),UNIT=SYSDA, DISP=(NEW,DELETE),
// DSORG=LS, LRECL=80, BLKSIZE=3200, RECFM=FB
//SYSSTIN DD SPACE=(TRK,(1,1)),UNIT=SYSDA, DISP=(NEW,DELETE),
// DSORG=PS, LRECL=137, BLKSIZE=4096, RECFM=VBA
//SYSPRINT DD SPACE=(CYL,(15,15)),UNIT=SYSDA, 
// DISP=(NEW,DELETE)
//SYSTSPRT DD SPACE=(CYL,(5,5)),UNIT=SYSDA, DISP=(NEW,DELETE),
// DSORG=PS, LRECL=137, BLKSIZE=4096, RECFM=VBA
//ALUIN  DD SPACE=(TRK,(1,1)),UNIT=SYSDA, DISP=(NEW,DELETE),
// DSORG=PS, LRECL=80, BLKSIZE=3200, RECFM=FB
//ALUPRINT DD SPACE=(CYL,(5,5)),UNIT=SYSDA, DISP=(NEW,DELETE)
//AJXPRINT DD SPACE=(CYL,(5,5)),UNIT=SYSDA, DISP=(NEW,DELETE)
/** RUNTIME ENABLEMENT CHOSEN
//SYSPROC DD DSN=AEX.QA1020.BMCCLIB, DISP=SHR
//AEXPRINT DD SYSOUT=*
//WORKLIST DD DISP=SHR,
// DD=ACM.DB2V10.V102.WLBASE01(MIGDB)
//***************************************************************************
//* UTILITY WORK DD STATEMENTS
```
d After you review the JCL, press END to return to the Execution JCL Processing Interface panel.

e From the Execution JCL Processing Interface panel, press Enter to submit the JCL and migrate your objects.

The result of execution is a worklist execution log (AEXPRINT).

f To review the results of execution in the worklist execution log (AEXPRINT), you can split your ISPF session or you can use your normal method to review SYSOUT. Figure 53 on page 113 shows the worklist execution log.
The last message in the worklist indicates that the work ID status is complete and that the worklist completed successfully. The worklist migrated the objects that you specified.

Figure 53: Execution results for a migrate-type worklist
**STATEMENTS**

- :*  STANDALONESTATS DB2STATSUPD TABLEALL
- *: NOHISTORY
- ** BMCCOPY
- ** BMCLoad NOUTILCOPY
- ** BMCFastload
- ** MAXSYSUT 20
- ** DYNUNLD
- ** DYNCOPY
- ** NOPARTCOPY
- ** COPYDDN (COPY01 )
- ** REGISTER (COPY01 )
- ** NOREGENIDENTITY
- ** SINGLEPHASE
- ** NOVALPROP
- ** BCMUNLOAD NOUNLDCOLL
- ** BMCREORG IBMCCOLL
- ** SYNCPPOINT 10
- ** NODBRMLIB
- ** ORDERBY
- ** UNLOADTABLE
- ** UNLOADEMPTY
- ** NOSTOPCOMMIT
- ** TABLEACCESS
- ** NOPARALLEL
- ** NOREORGONLINE

---

- ** KEYCARD
- ** REORGALL
- ** REORGREF
- ** DYNREORG
- ** CLONEDATA

---

- ** TIME 000050 '2013-01-15-14.32.21.382780' 57154342
- ** SSID 000100 DEGA 14266081
- ** JCLP 000150 MIGGER MSSID DEGA 62418581
- ** JCLP IGNORED IN THIS RUN. SKIPPING WORKLIST RECORDS UNTIL NEXT COMMAND
- ** WKID 000200 RDACRJ,MIGDB 90835983

---

- ** THE FOLLOWING MESSAGES ARE WARNINGS ONLY

---

- **BMC45339W TAPES CANNOT BE STACKED AND MULTIPLE TAPE DRIVES MAY BE REQUIRED IF MULTITASKING IS USED TO UNLOAD AND LOAD

---

**STATEMENTS**

**STATEMENTS**
PARTITIONED TABLE SPACES. TO DISABLE MULTITASKING AND TO
ENABLE TAPE STACKING, ADD THE NOMTASK KEYWORD TO THE ALUIN
INPUT STREAM.

******************************************************************************
*** END OF WARNING MESSAGES - WARNINGS WERE ISSUED                     ***
******************************************************************************
*** ORIGINATOR OF WORKLIST WAS WORKID RDACRJ.MIGDB ON SSID DEGA ***
: MIGR 000250 PHASE-1 START OF SECTION TO BE RUN ON SENDING SYSTEM 21838478
  **** B M C  E X E C U T I O N  F O R  D B 2
  **** V E R S I O N: V11.01.00
  (C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 15.05.37 ON SYSID DB2B WORKLIST EXECUTION
LOG PAGE 4

LINE PREFIX : DENOTES WORKLIST INPUT

---SYNC 000300 START OF EXECUTION SYNCPOINT 02891110
- JCLP 000350 BMCD IPPARTS 4 18591912
BMCD 000400
  UNLOAD UNLOADDN (SYSR)
  ACTIVE (YES,NO)
  ON FAILURE ALL TERMINATE UTILITY
  DELETENAMES YES
  DISCARDS 1
  NULLTYPE TI NULLCHAR ?
  FORMAT BMCLOAD
  SELECT *
  FROM R1BASIC.T_TO1AICP
  ORDER BY
    COLSI_2
    , COLVC_8
  ;
  OUTPUT SYSR
  UNIT SYSDA
  DSNAME 'RDACRJ.DEGA.MIGDB.SR000001.P&PART'
  
BMC16898I DEBUG: CONNECTED WITH USER: RDACRJ
BMC16898I DEBUG: ADUUMAIN 0

***** B M C  U N L O A D  P L U S  F O R  D B 2
V10R2.00 *****

(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.
UNLOAD PLUS TECHNOLOGY IS PROTECTED BY U.S. PATENT

NUMBER 5,860,069

BMC50001I UTILITY EXECUTION STARTING 1/15/2013 15:05:41 ...
BMC50002I UTILITY ID = 'MIGDB'. DB2 SUBSYSTEM ID = 'DEGA'. OPTION MODULE = 'ADU$OPTS'.
BMC50471I z/OS 1.13.0,PID=HBB7780,DFSMS FOR Z/OS=1.13.0,DB2=10.1.0
BMC50471I REGION=OM,BELLOW 16M=8788K,ABOVE 16M=1416920K,IEFUSI=NO,CPU$=3
BMC50471I MEMLIMIT=17592186040320M,AVAILABLE=17592186040314M,MEMLIMIT SET BY:JCL
BMC50471I UNLOAD PLUS FOR DB2--V10.02.00
BMC50471I NO MAINTENANCE TO REPORT
BMC50471I DB2 UTILITIES COMMON CODE--V10.02.00
BMC50471I NO MAINTENANCE TO REPORT
BMC50471I SOLUTION COMMON CODE--V11.01.00

Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  115
# Achieving your goals with ALTER and CHANGE MANAGER

## ALTER and CHANGE MANAGER for DB2 User Guide

```
BMC50471  NO MAINTENANCE TO REPORT
BMC50471  BMCSORT ENGINE--V02.03.01

****apses BMC EXECUTION FOR DB2
****
**** VERSION: V11.01.00
(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 15.06.03 ON SYSID DB2B
WORKLIST EXECUTION
LOG
PAGE  5

LINE PREFIX : DENOTES WORKLIST INPUT

STATEMENTS

BMC50471 MAINT: BPJ0195 BPJ0198 BPJ0209 BPJ0250 BPJ0263 BPJ0269 BPJ0276
BPJ0279 BPJ0288 BPJ0308 BPJ0343 BPJ0357
BPJ0471 BPJ0361 BPJ0374 BPJ0390 BPJ0404 BPJ0473 BPJ0503 BPJ0521
BPJ0529 BPJ0591
BMC50471 BMC STATS API--V11.01.00
BMC50471 NO MAINTENANCE TO REPORT
BMC50471 EXTENDED BUFFER MANAGER--V06.01.00
BMC50471 NO MAINTENANCE TO REPORT
BMC50471 ACFORTSS=YES                            FORCE_AT=(START,
3)    SHRLEVEL=REFERENCE
BMC50471 ANALYZE=(DB2STATS,NOLIMIT) SMAX=16
FORCERPT=NO
BMC50471 CENTURY=(1950,2049) SMCORE=(OK,OK)
FORCENONE
BMC50471 CHANGE_CONSISTENT=NO SORTNUM=32
HISTORY=YES      
BMC50471 CHANGEQUIRES=NO
IBUFFS=25
BMC50471 CMAX=16 SQLDELAY=3
INLINE=NO
BMC50471 CMRATIO=50 TAPEDISP=DELETE
LOADDECP=NO
BMC50471 CONRULES=BMC          TASKMAX=200%
LOCKROW=NO
BMC50471 CURRENTDEGREE=NONE UBUFFS=25
MAXP=5
BMC50471 DFLFILES=YES
MSGLEVEL=1
BMC50471 DRNDelay=1
NULLCHAR=X'6F'
BMC50471 DRNRETRY=255 UNLOADActive=(YES,NO)
BMC50471 T1
NULLTYPE=T1
BMC50471 DRNWAIT=NONE UNLOAD=(SYSREC,SYSRED)
OPNDB2ID=YES USELRECL=NO
BMC50471 DSPLOCKS=DRNFLA PLAN=ADUQA
BMC50471 EXCLDUMP=(X37,X22,X06) UXSTATE=SUP
RECFM=AUTO
BMC50471 FILEREFDN=SYSREF WORKUNIT=SYSALLDA
ROWSETSZ=100 ZIIP=ENABLED
BMC50471 FILL=NO
SDUMP=YES
BMC50471 TAPES=NONE
BMC504701 OUTPUT = SYSREC
SYSRED
BMC504701 UNIT = SYSALLDA SYSREF
SYSSALDA
BMC504701 VOLCNT = 25 25
```
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  117
Achieving your goals with ALTER and CHANGE MANAGER

BMC50471I LOCAL TIME LENGTH = 0
BMC50471I DECIMAL POINT = PERIOD
BMC50471I DECIMAL ARITHMETIC = 15
BMC50471I DELIMITER = DEFAULT
BMC50471I SQL DELIMITER = DEFAULT
BMC50471I ENCODING SCHEME = EBCDIC
BMC50471I APPL. ENCODING SCHEME = EBCDIC
BMC50471I MIXED = NO
BMC50471I EBCDIC CCSID = (37,65534,65534)
BMC50471I ASCII CCSID = (819,65534,65534)
BMC50471I UNICODE CCSID = (367,1208,1200)

BMC50028I DB2 MODE = NFM
BMC50471I BMC_BMCUTIL = 'BMCUTIL.CMN_BMCUTIL'
BMC50471I BMC_BMCSYNC = 'BMCUTIL.CMN_BMCSYNC'
BMC50471I BMC_BMCHIST = 'BMCUTIL.CMN_BMCHIST'
BMC50471I BMC_BMCXCOPY = 'BMCUTIL.CMN_BMCXCOPY'

BMC50102I UNLOAD UNLOADDN (SYSR)
BMC50102I ACTIVE (YES,NO)
BMC50102I ON FAILURE ALL TERMINATE UTILITY
BMC50102I DISCARDS 1
BMC50102I NULLTYPE T1 NULLCHAR ?
BMC50102I FORMAT BMCLOAD

** B M C E X E C U T I O N F O R D B 2 **

BMC50445I UNLOAD PLUS DYNAMIC FILE ALLOCATION REPORT
BMC50446I                  UNIT
BMC50447I DDNAME   DSNAME             DATACLAS MGMTCLAS
BMC50447I STORCLAS PRI      SEC      PRI       SEC
BMC50448I SYS00006 RDACRJ.DEGA.MIGDB.SR000001.P001
BMC50448I SYS00007 RDACRJ.DEGA.MIGDB.SR000001.P002

118 ALTER and CHANGE MANAGER for DB2 User Guide
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  119
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
::SYNC 000450 UNLOAD TB RIBASIC.T_T01AICP COMPLETE 97412899
::SYNC 000500 END OF UNLOAD TABLES SECTION 36723126
::---------------------------------------------------------------------
::SETS 000550 SET CURRENT SQLID = 'RDACRJ' 34458304
::---------------------------------------------------------------
::DSNT400I SQLCODE = 000, SUCCESSFUL EXECUTION
::SQL 000600 CREATE DATABASE RJBASIC
::                    STOGROUP SYSDEFLT
::                    BUFFERPOOL BP8K0
::                    LOCKSIZE   ANY
::                    LOCKMAX    10
::                    CLOSE      NO
::                    SEGSIZE    0
::                    PRIQTY     440
::                    SECQTY     220
::                    PCTFREE    0
::                    PRIQTY     440
::                    SECQTY     220
::                    PCTFREE    0
::                    PRIQTY     440
::                    SECQTY     220
::                    PCTFREE    0
::                    PRIQTY     440
::                    SECQTY     220
::                    PCTFREE    0
::                    PRIQTY     440
::                    SECQTY     220
::                    PCTFREE    0
::---------------------------------------------------------------------
::SETS 000750 SET CURRENT SQLID = 'RDACRJ' 34589376
::---------------------------------------------------------------------
::DSNT400I SQLCODE = 000, SUCCESSFUL EXECUTION
::SETS 000800 CREATE TABLESPACE T01AICP IN RJBASIC
::                   NUMPARTS 4
::                   )
::                   )
::                   )
::                   )
::---------------------------------------------------------------------

**** B M C   E X E C U T I O N   F O R   D B 2
****

V E R S I O N: V11.01.00
(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 15.06.03 ON SYSID DB2B WORKLIST EXECUTION
LOG PAGE 9
DSNT400I SQLCODE = 000, SUCCESSFUL EXECUTION
:: SETS 000850 SET CURRENT SQLID = USER 12025904
DSNT400I SQLCODE = 000, SUCCESSFUL EXECUTION
:: SYNC 000900 END OF CREATE TABLESPACE SECTION 21939128

*******************************************************************************
:: SQL 000950 CREATE TABLE RJBASIC.T_T01AICP
:: (  
::  COLC_1  
::   CHAR(20) NOT NULL WITH DEFAULT  
::  , COLSI_2  
::   SMALLINT NOT NULL WITH DEFAULT  
::  , COLIN_3  
::   INTEGER NOT NULL WITH DEFAULT  
::  , COLSI_4  
::   SMALLINT NOT NULL WITH DEFAULT  
::  , COLDC_5  
::   DECIMAL(6,2) NOT NULL WITH DEFAULT  
::  , COLDC_6  
::   DECIMAL(9,0) NOT NULL WITH DEFAULT  
::  , COLDC_7  
::   DECIMAL(5,2) NOT NULL WITH DEFAULT  
::  , COLVC_8  
::   VARCHAR(15) NOT NULL WITH DEFAULT  
::  , COLF1_9  
::   REAL NOT NULL WITH DEFAULT  
::  , COLF2_10  
::   FLOAT NOT NULL WITH DEFAULT  
::  , COLVC_11  
::   VARCHAR(15) NOT NULL WITH DEFAULT  
::  , COLDT_12  
::   DATE NOT NULL WITH DEFAULT  
::  , COLTS_13  
::   TIMESTAMP NOT NULL WITH DEFAULT  
::  , COLC_14  
::   CHAR(100) NOT NULL WITH DEFAULT  
:: )  
::                  CCSID EBCDIC 01354097
:: IN RJBASIC.T01AICP
::  
::  DSNT400I SQLCODE = 000, SUCCESSFUL EXECUTION
::  - SQL 001000 COMMENT ON TABLE RJBASIC.T_T01AICP  
::    ( IS 'T01- FACTS DO NOT CEASE TO EXIST BECAUSE THEY A  
::      RE IGNORED'  
::  DSNT400I SQLCODE = 000, SUCCESSFUL EXECUTION
::  - SYNC 001050 END OF TABLE SECTION 75666657
:: *******************************************************************************
:: SQL 001100 CREATE INDEX
::  RJBASIC.I_T01AICP1
:: ON RJBASIC.T_T01AICP
::  
::  **** B M C   E X E C U T I O N   F O R   D B 2
::  **** V E R S I O N:  V11.01.00
:: (C) COPYRIGHT 1991 - 2013 BMC SOFTWARE.
:: INC.
:: 2013-01-15 15.06.05 ON SYSID DB2B WORKLIST EXECUTION
::  
:: LINE PREFIX : DENOTES WORKLIST INPUT
::  
::  DSNT400I SQLCODE = 000, SUCCESSFUL EXECUTION
::  - SQL 001100 CREATE INDEX
::  RJBASIC.I_T01AICP1
::  ON RJBASIC.T_T01AICP
::  
::  Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  121
DSNT400I SQLCODE = 000, SUCCESSFUL EXECUTION

**** B M C   E X E C U T I O N   F O R   D B 2

(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 15.06.08 ON SYSID DB2B WORKLIST EXECUTION LOG PAGE 11

LINE PREFIX : DENOTES WORKLIST INPUT

BMC16511I -JCLP IGNORED IN THIS RUN. SKIPPING WORKLIST RECORDS UNTIL NEXT COMMAND

BBMCL 001250

LOAD DATA REPLACE

INDSN('RDACRJ.DEGA.MIGDB.SR000001.P001'

'RDACRJ.DEGA.MIGDB.SR000001.P002'

'RDACRJ.DEGA.MIGDB.SR000001.P003'

'RDACRJ.DEGA.MIGDB.SR000001.P004'

)

ERRRDN SYSER001
DISCARDS 1
DISCARDON SYSDS001
DELETEFILES NO
ORDER YES
FORMAT BMCUNLOAD
WORKDDN SYSDT001
LOADDDN SORTP
ENFORCE NO
REDEFINE NO
UNIQUEINTO YES
INTO TABLE RJBASIC.T_T01AICP
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface    123
Achieving your goals with ALTER and CHANGE MANAGER

124  ALTER and CHANGE MANAGER for DB2 User Guide
### Worklist Execution Log

**Date and Time:** 2013-01-15 15.06.47
**System ID:** DB2B

---

**Line Prefix:** Denotes Worklist Input

<table>
<thead>
<tr>
<th>Statement</th>
<th>Value 1</th>
<th>Value 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVGVOLSP</td>
<td>((30000,TRK),(30000,TRK))</td>
<td>((30000,TRK), (30000,TRK))</td>
</tr>
<tr>
<td>DSNTYPE</td>
<td>(NONE,NONE)</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>DATACLAS</td>
<td>(NONE,NONE)</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>MGMTCLAS</td>
<td>(NONE,NONE)</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>STORCLAS</td>
<td>(NONE,NONE)</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>THRESHLD</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MAXEXTSZ</td>
<td>((0,K),(0,K))</td>
<td>(0,K)</td>
</tr>
<tr>
<td>EXPDTR</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>RETPD</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>GDGLIMIT</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>GDGEMPTY</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>GDGSCRAT</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>DDTYPE</td>
<td>ERROR DISCARD</td>
<td>LOCPFCPY</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>IFALLOC</td>
<td>USE</td>
<td>USE</td>
</tr>
<tr>
<td>SMS</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>SMSUNIT</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>SIZEPCT</td>
<td>(100,100)</td>
<td>(100,100)</td>
</tr>
<tr>
<td>UNIT</td>
<td>(SYSALLDA,SYSALLDA)</td>
<td>(SYSALLDA,SYSALLDA)</td>
</tr>
<tr>
<td>UNITCNT</td>
<td>(0,0)</td>
<td>(0,0)</td>
</tr>
<tr>
<td>VOLCNT</td>
<td>(25,25)</td>
<td>(25,25)</td>
</tr>
</tbody>
</table>

---

**Chapter 2** Overview of the ALTER and CHANGE MANAGER interface 125
(NONE, NONE) STORCLAS = (NONE, NONE)
(NONE, NONE) THRESHLD = 0
0
BMC50470I MAXEXTSZ = ((0,K),(0,K))
(0,K) ((0,K),(0,K))
BMC50470I EXPDT =
BMC50470I RETPD =
BMC50470I GDGLIMIT = N/A
A
BMC50470I GDGEMPTY = N/A
A
BMC50470I GDGSCRAT = N/A
A
BMC50470I DDTYPE = LOCBFCPY
REMPFCPY
BMC50470I ACTIVE = NO
NO
BMC50470I IFALLOC = USE
USE
BMC50470I SMS = NO
NO
BMC50470I SMSUNIT = NO
NO
BMC50470I SIZEPCT = (100,100)
(100,100)
BMC50470I UNIT = (SYSALLDA, SYSALLDA)
(SYSALLDA, SYSALLDA)
BMC50470I UNITCNT = (0,0)
(0,0)
BMC50470I VOLCNT = (25,25)
(25,25)
BMC50470I AVGVOLSP = ((30000,TRK),(30000,TRK))
((30000,TRK),(30000,TRK))
BMC50470I DSNTYPE = (NONE, NONE)
(NONE, NONE)
BMC50470I DATACLAS = (NONE, NONE)
(NONE, NONE)
BMC50470I MGMTCLAS = (NONE, NONE)
(NONE, NONE)
BMC50470I STORCLAS = (NONE, NONE)
(NONE, NONE)
BMC50470I THRESHLD = 0
0
BMC50470I MAXEXTSZ = ((0,K),(0,K))
(0,K) ((0,K),(0,K))
BMC50470I EXPDT =

**** BMC EXECUTION FOR DB2

**** VERSION: V11.01.00
(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 15.06.47 ON SYSID DB2B
WORKLIST EXECUTION

PAGE 14

------------------------------------------------------------------------------------------
LINE PREFIX : DENOTES WORKLIST INPUT
BMC50470I GDGEMPTY = NO
BMC50470I GDGSCRAT = NO
BMC50470I DDTYPE = SYSMAP
BMC50470I ACTIVE = NO
BMC50470I IFALLOC = USE
BMC50470I SMS = NO
BMC50470I SMSUNIT = NO
BMC50470I SIZEPCT = (100,100)
BMC50470I UNIT = (SYSALLDA,SYSALLDA)
BMC50470I UNITCNT = (0,0)
BMC50470I VOLCNT = (25,25)
BMC50470I AVGVOLSP = ((30000,TRK),(30000,TRK))
BMC50470I DSNTYPE = (NONE,NONE)
BMC50470I DATACLAS = (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)
BMC50470I THRESHLD = 0
BMC50470I MAXEXTSZ = ((0,K),(0,K))
BMC50470I EXPDT = N/A
BMC50470I RETPD = N/A
BMC50470I GDGLIMIT = 5
BMC50470I GDGEMPTY = N/A
BMC50470I GDGSCRAT = NO
BMC50470I DDTYPE = REMPLCPY
BMC50470I ACTIVE = NO
BMC50470I IFALLOC = USE
BMC50470I SMS = NO
BMC50470I SMSUNIT = NO
BMC50470I SIZEPCT = (100,100)
BMC50470I UNIT = (SYSALLDA,SYSALLDA)
BMC50470I UNITCNT = (0,0)
BMC50470I VOLCNT = (25,25)
BMC50470I AVGVOLSP = ((30000,TRK),(30000,TRK))
BMC50470I DSNTYPE = (NONE,NONE)
BMC50470I DATACLAS = (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)
BMC50470I THRESHLD = 0
BMC50470I MAXEXTSZ = ((0,K),(0,K))

Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  127
<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MGMTCLAS</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>STORCLAS</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>THRESHLD</td>
<td>0</td>
</tr>
<tr>
<td>MAXEXTSZ</td>
<td>((0,K),(0,K))</td>
</tr>
<tr>
<td>EXPDT</td>
<td>5</td>
</tr>
<tr>
<td>GDGEMPTY</td>
<td>NO</td>
</tr>
<tr>
<td>GDGSCRAT</td>
<td>NO</td>
</tr>
<tr>
<td>DDTYPE</td>
<td>LOCBXCPY</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>NO</td>
</tr>
<tr>
<td>SMS</td>
<td>NO</td>
</tr>
<tr>
<td>SMSUNIT</td>
<td>NO</td>
</tr>
<tr>
<td>SIZEPCT</td>
<td>(100,100)</td>
</tr>
<tr>
<td>UNIT</td>
<td>(SYSALLDA,SYSALLDA)</td>
</tr>
<tr>
<td>UNITCNT</td>
<td>(0,0)</td>
</tr>
<tr>
<td>VOLCNT</td>
<td>(25,25)</td>
</tr>
<tr>
<td>AVGVOLSP</td>
<td>((30000,TRK),(30000,TRK))</td>
</tr>
<tr>
<td>DSNTYPE</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>DATACLAS</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>MGMTCLAS</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>STORCLAS</td>
<td>(NONE,NONE)</td>
</tr>
<tr>
<td>THRESHLD</td>
<td>0</td>
</tr>
<tr>
<td>MAXEXTSZ</td>
<td>((0,K),(0,K))</td>
</tr>
<tr>
<td>EXPDT</td>
<td>5</td>
</tr>
<tr>
<td>RETPD</td>
<td></td>
</tr>
<tr>
<td>GDGLIMIT</td>
<td>5</td>
</tr>
<tr>
<td>GDGEMPTY</td>
<td>NO</td>
</tr>
<tr>
<td>GDGSCRAT</td>
<td>NO</td>
</tr>
</tbody>
</table>

---

**Achieving your goals with ALTER and CHANGE MANAGER**

128. *ALTER and CHANGE MANAGER for DB2 User Guide*
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface    129
Achieving your goals with ALTER and CHANGE MANAGER

ALTER and CHANGE MANAGER for DB2 User Guide
Chapter 2 Overview of the ALTER and CHANGE MANAGER interface

131

**BMC EXECUTION FOR DB2**

****

**** VERSION: V11.01.00

(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 15.06.47 ON SYSID DB2B

**STATEMENTS**

**LINE PREFIX :** DENOTES WORKLIST INPUT
Achieving your goals with ALTER and CHANGE MANAGER

---

BMC50477I 0: PARTITION = 2, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00022
BMC50482I 0: LOAD COMPLETE. ELAPSED TIME = 00:00:00 DSN =
'DEGACAT.DSNDBD.RJBASIC.T01AICP.I00001.A002'
BMC50477I 0: PARTITION = 2, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00023
BMC51474I LOAD STATISTICS: 0 ROWS LOADED INTO TABLE SPACE 'RJBASIC.T01AICP' PARTITION 2
BMC50482I 0: BUILD COMPLETE. ELAPSED TIME = 00:00:02 DSN =
'DEGACAT.DSNDBD.RJBASIC.IRT01AICP.I00001.A003'
BMC50477I 0: PARTITION = 3, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00024
BMC50482I 0: LOAD COMPLETE. ELAPSED TIME = 00:00:00 DSN =
'DEGACAT.DSNDBD.RJBASIC.T01AICP.I00001.A004'
BMC50477I 0: PARTITION = 3, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00025
BMC51474I LOAD STATISTICS: 0 ROWS LOADED INTO TABLE SPACE 'RJBASIC.T01AICP' PARTITION 3
BMC51476I BUILD STATISTICS: 2 KEYS LOADED INTO INDEX 'RJBASIC.I_T01AICP1'
BMC51475I LOAD STATISTICS: 2 ROWS LOADED INTO TABLE SPACE 'RJBASIC.T01AICP'
BMC50041I 0: ZIIP NOT ENABLED (0) USING XBM SUBSYSTEM XBMB
BMC50004I LOAD PHASE COMPLETE. ELAPSED TIME = 00:00:10
BMC50511I DB2 OBJECT STATISTICS
BMC50515I TABLESPACE RJBASIC.T01AICP PARTS = 4 TABLES = 1 SEGSIZE = 0 DSSIZE = 0 PAGESIZE = 4K
BMC50516I PART NACTIVE CARD EXTENTS DBCARD PCOMP KSAVED PSAVED
BMC50517I 1 2 1 1 0 0 0 0
BMC50517I 2 1 0 1 0 0 0 0
BMC50517I 3 1 0 1 0 0 0 0
BMC50517I 4 2 1 1 0 0 0 0
BMC50518I TABLE RJBASIC.T_T01AICP
BMC50519I ROWAVG NPAGES CARD PCTPAGES
BMC50520I 190 2 2 33
BMC50521I INDEX RJBASIC.I_T01AICP1
BMC50526I FIRST KEY COLUMN = COLSI_2
BMC50527I CLUSTER = Y UNIQUE = D COMPRESS = N PAGESIZE = 4K KEYLEN = 19 COLCOUNT = 2
BMC50528I PART NACTIVE CARD EXTENTS LEVELS
BMC50529I 1 5 1 1 2
BMC50529I 2 5 0 1 2
BMC50529I 3 5 0 1 2
BMC50529I 4 5 1 1 2
BMC50290I DB2 REAL-TIME-STATISTICS -RESET STATS- FUNCTION FOR LOAD UTILITY SUCCESSFUL FOR ALL OBJECTS
BMC50387W IMAGE COPY REQUIRED. TABLE SPACE STATE SET TO "COPY PENDING"
BMC50004I UTILTERM PHASE COMPLETE. ELAPSED TIME = 00:00:00
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 4
::SYNC 001300 LOAD TB RJBASIC.T_T01AICP COMPLETE 26912713
::SYNC 001350 END OF LOAD TABLES SECTION 61548774
::BMCC 001400
::OUTPUT DCPYLOC
::UNIT SYSDA

**** BMC EXECUTION FOR DB2

**** VERSION: V11.01.00
(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 15.06.47 ON SYSID DB2B
WORKLIST EXECUTION
LOG PAGE 19

STATEMENTS

:: DSNAME &UID.MIGDB.&TS.P&PART.LP000001

---

132 ALTER and CHANGE MANAGER for DB2 User Guide
BMC16815I PARAMETERS: DEGA,MIGDB
BMC16822I Control PASSED TO 'ACPMAIN'
BMC16898I DEBUG: CONNECTED WITH USER: RDACRJ
BMC16970I CURRENT SQLID REESTABLISHED VIA: SET CURRENT SQLID = USER
BMC16898I DEBUG: ACPMAIN 0

BMC30101I COPY PLUS FOR DB2 V11.1.00
BMC47491I COPYRIGHT BMC SOFTWARE INC. 1991-2013
BMC47487I COPY PLUS TECHNOLOGY IS PROTECTED BY U.S. PATENTS 7,133,884 AND 7,769,718
BMC30001I UTILITY EXECUTION STARTING 01/15/2013 15:06:48...
BMC30002I UTILITY ID = MIGDB. DB2 SUBSYSTEM ID = DEGA.
BMC30008I RESTART PARM =
BMC180107I COPY PLUS STEP STEP1 MAINTASK TCB X'008961A0'
BMC30145I MAINT: NO COPY PLUS PTFS APPLIED
BMC30101I SOLUTION COMMON CODE V11.1.00
BMC30145I MAINT: NO SCC PTFS APPLIED
BMC30101I USING STANDARD DB2 SECURITY, EXIT(12)
BMC30594I EBCDIC CCSID = 37
BMC30101I DATABASE ADMINISTRATION FOR DB2 SOLUTION V10.1.00
BMC30101I RECOVERY MANAGEMENT FOR DB2 V11.1.00

**** B M C   E X E C U T I O N   F O R   D B 2
****
**** V E R S I O N:  V11.01.00
(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 15.07.06 ON SYSID DB2B
LOG  PAGE 20

STATEMENTS

BMC30101I SUBTASK 1 COMPLETE FOR SPACE RJBASIC.T01AIKP DSNUM 0, RC = 0
BMC3010I
BMC16066I PROCESSING COMMAND COPY ON RJBASIC.T01AICP
BMC30005I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
  ::SYNC 001450 COPY OF MULTIPLE TABLESPACES COMPLETE                  32069875
  ::SYNC 001500 END OF COPY SECTION                                        93709873
  ####################################################################
  ::RNST 001550
               RUNSTATS TABLESPACE RJBASIC.T01AICP
                        HISTORY NONE
                        INDEX (ALL)
                        KEYCARD
                        TABLE ALL
                                              05716088
BMC16815I PARAMETERS: DEGA,MIGDB
BMC16822I Control PASSED TO 'DSNUTILB'
BMC16898I DEBUG: CONNECTED WITH USER: RDACRJ
BMC16970I CURRENT SOLID REESTABLISHED VIA: SET CURRENT SOLID = USER
BMC16898I DEBUG: DSNUTILB 0

DSNU000I 015 15:07:08.02 DSNUGUTC - OUTPUT START FOR UTILITY, UTILID = MIGDB
DSNU1044I 015 15:07:08.63 DSNUGTIS - PROCESSING SYSIN AS EBCDIC
DSNU050I 015 15:07:08.67 DSNUGUTC - RUNSTATS TABLESPACE RJBASIC.T01AICP HISTORY NONE
INDEX(ALL) KEYCARD TABLE
 ALL
DSNU610I *DEGA 015 15:07:10.89 DSNUSUPUTP - SYSTABLEPART CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:10.90 DSNUSUPUT - SYSTABSTATS CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:11.05 DSNUSUPC - SYSCOLSTATS CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:11.15 DSNUSUTB - SYSTABLES CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:11.22 DSNUSUCO - SYSCOLUMNS CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:11.24 DSNUSUTS - SYSTABLESPACE CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:11.58 DSNUSUIP - SYSINDEXPART CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:11.60 DSNUSUPI - SYSINDEXSTATS CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:11.62 DSNUSUPC - SYSCOLSTATS CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:11.63 DSNUSUPD - SYSCOLDISTSTATS CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:11.68 DSNUSUCO - SYSCOLUMNS CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:11.74 DSNUSUX - SYSINDEXES CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU610I *DEGA 015 15:07:11.81 DSNUSUCD - SYSCOLDIST CATALOG UPDATE FOR
 RJBASIC.T01AICP SUCCESSFUL
DSNU620I *DEGA 015 15:07:11.82 DSNUSEOF - RUNSTATS CATALOG TIMESTAMP =
2013-01-15 15:07.08.802484
DSNU010I 015 15:07:11.99 DSNUGBAC - UTILITY EXECUTION COMPLETE, HIGHEST RETURN CODE=0
  ::SYNC 001600 STATS FOR TS RJBASIC.T01AICP COMPLETE                  25273345
  ::SYNC 001650 END OF TABLESPACE STATISTICS SECTION                    41697102
  ****  BMC EXECUTION FOR DB2  
****  V E R S I O N:  V11.01.00  
(C) COPYRIGHT 1991 - 2013 BMC SOFTWARE, INC.

2013-01-15 15:07.12 ON SYSID DB2B WORKLIST EXECUTION
LOG PAGE 21

--------------------------------------------------------------------------

LINE PREFIX : DENOTES WORKLIST INPUT

Achieving your goals with ALTER and CHANGE MANAGER
Press END to return to the Migrate DB2 Objects Using Specification panel.

Press END to return to the Task List Menu panel.

Press END to return to the CHANGE MANAGER Main Menu.

This example shows that with CHANGE MANAGER you have the ability to quickly and accurately specify objects that you want to migrate to another environment, perform an in-depth analysis of the objects selected to determine their effect on your subsystem, and execute SQL statements, utilities, and other commands.
Setting up ALTER and CHANGE MANAGER

This section describes how to set options for how ALTER and CHANGE MANAGER operate, restrict access to the components of the products, and maintain the tables used by the products.

Before you set up the ALTER or CHANGE MANAGER product for the ISPF interface, ensure that you have completed the tasks necessary to install the products. For information, see:

- BMC Products and Solutions for DB2 Customization Guide
- Installation System Quick Start
- Installation System Reference Manual

In addition, configure a TSO region size of 8 MB or greater, for each user who signs on.

Using options to control your environment

For your production requirements, you should reevaluate the option settings for ALTER or CHANGE MANAGER. For most situations, the installation default values are sufficient. However, you might want to customize the operating environment and panels.

ALTER or CHANGE MANAGER uses default, user, and product options to define the operating environment and to specify how the product’s components work. The options also contain default values for data set names and allocations, job control language (JCL) generation information, and component plan names.

These options provide you with the ability to:

- Tailor the interface
Set up defaults for generating job statements

Specify defaults for parameters, names, and prefixes for allocated data sets

Specify processing options for the Analysis, JCL Generation, Execution, Import, Baseline, and CM/PILOT components

Typically, the person who installs ALTER or CHANGE MANAGER sets default values for user options. For information about establishing installation option values at installation, see:

- BMC Products and Solutions for DB2 Customization Guide
- Installation System Quick Start
- Installation System Reference Manual

Using the installation options

The default operating environment is controlled by a number of option values that are defined in the installation options module.

The components of ALTER or CHANGE MANAGER use the global values that are stored in the installation options module to determine how to process information. The Installation System generates the installation options module when you install ALTER or CHANGE MANAGER. The module contains an assembly-language program with an options macro.

You can customize ALTER or CHANGE MANAGER for all users by editing the default values in the installation options module. The default name of the module is ACMDOP1. The source of the installation option modules is located in the HLQ.UBMCCNTL data set. HLQ identifies the high-level qualifier that you specify when you install the products.

Note
ALTER or CHANGE MANAGER uses plan names directly. If you need to specify different plan name values for each DB2 subsystem, you must have multiple installation option modules.

To display the installation options

1. From the Main Menu panel, enter DOPTS on the Command line to display the installation options that are in effect for the current CHANGE MANAGER session.
Using the user options

The first time that you run ALTER or CHANGE MANAGER, the product copies the values from the installation options module into your ISPF profile.

The values in this profile are referred to as your user options. The product uses these user options to generate JCL and to generate keywords for an input stream for each user’s subsystem. The product uses the following input streams:

- AEXIN input stream, which is used by the Execution component
- AEXPIN, which is used by the Execution component for the worklist parallelism feature in the Database Administration solution
- ALUIN, which is used by the Analysis, Baseline, Baseline Report, Compare, CM/Pilot and Import components
- AJXIN and AJXPOFIN, which are used by the Batch Execution JCL Generation component

Note
AJXPOFIN is also used by Batch Analysis JCL Generation.

You can refresh the values in your user options by editing and reassembling the installation options module.

Storing values in the ISPF profile

The ISPF profile resides in the data set member prdxPROF, where prdx is the value of the application ID in the BMCDB2 CLIST. To define or modify the values in this profile, you can use the Options panels of ALTER or CHANGE MANAGER, JCL Generation, or CM/PILOT. You can use literal characters or symbolic variables to specify the values on the Options panels. (For more information about symbolic variables, see the ALTER and CHANGE MANAGER for DB2 Reference Manual.)

1. To save your user options in your profile, exit the product.

   Note
   If your ISPF session abnormally ends (abends), the user options that you modified are not saved.

To refresh an option value in all existing ISPF profiles

You can refresh the values in your user options by editing and reassembling the installation options module.
Type a comma and an R after the option value and then enclose the value in parentheses, as shown in the following example:

\[ \text{DBCS=(N,R), } * \]

**Note**

Do not remove the comma after the right parenthesis or the continuation character (*) in column 72, except for the last option value.

The next time that you run the product, the new global value replaces the old local value in the user options. You can modify the local value through the options panels. If you need to change the installation options after installation, you must reassemble the installation options module.

For more information about refreshing user options, see:

- **Installation System Reference Manual**
- **Installation System Quick Start**
- **BMC Products and Solutions for DB2 Customization Guide**

### Using the product options

The POFDS keyword (in the installation options module) specifies an 80-character sequential file. This product options file (POF) is built during product installation and contains parameters and values for the JCL Generation, ALTER, and CHANGE MANAGER options. The file is located in the HLQ.UBMCCNTL data set. The POF does not require assembly and linkage and does not need to reside in an APF-authorized data set.

The Batch Execution JCL Generation component uses the AJXIN keyword, POFDS, to specify the POF data set. Batch Execution JCL Generation can use an input stream with the ddname of AJXPOFIN to override keywords in the data set that the POFDS keyword specifies. For more information, see **ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2**.

When you install the products, only one POF is created. This initial POF is initialized and populated with the default ISPF variables and values from the installation panels. This POF is shared among several products, if those products are installed at the same time.

In addition, the Installation System uses the same application ID (or profile) for the products in the BMCDB2 CLIST. This single application ID enables the JCL Generation options to be shared with other products, such as the CATALOG
MANAGER product. Thus, when you specify an option for generating JCL in one product, your selection applies to all of the products.

**Note**
Although BMC recommends that you use a single application ID, you can choose individual product application IDs on the BMCDB2PR panel.

JCL Generation also handles user POFs, which are POFs that can be written from the ISPF variables that are set in the Front End component of ALTER or CHANGE MANAGER or edited. You can use a user POF to reset all of the options that you will use in the current session to create JCL. You can also use the user POFs to set options for different sets of applications, particularly if the applications have different naming standards.

**Note**
If a POF keyword in your user POF uses a library from an earlier version of the product, update the keyword to use a library for the most recently installed version of the product. For example, assume that the value of the BMC_COPY_LOAD keyword is BMC1010.ACM.D10.LOAD and then you installed version 11.2.00 of the product. Update the value of the keyword to a version 11.2.00 LINK library (BMC1120.ACM.D11.BMCLINK).

- The first time that the product is invoked, all of the values in the ISPF profile are set to the values that are in the initial POF. If a POF is not specified, default values are assigned to the variables in the profile.

- If the POFDATE parameter in the initial POF is greater than the value of the POF date that is stored in the ISPF profile, the values in the POF that are marked with refresh,(R) are used to reset the ISPF variables.

- If you specify a new initial POF in the POFDS installation option, the values in the POF that are marked with refresh,(R) are used to reset the ISPF variables. The value of the POFDATE keyword in the new initial POF is saved in the ISPF profile.

For more information about POFs, see the *ALTER and CHANGE MANAGER for DB2 Reference Manual*.

**Product option overrides**

You can override the user options for an ALTER or CHANGE MANAGER component by using the Override Options panels for that component.

For example, you can override the product options for JCL Generation by using the Override Options panels for Execution. (You can also override the options that are
related to dynamic allocation for JCL Generation in the same panels for Analysis.) The override options remain in effect only for the current ALTER or CHANGE MANAGER session.

The Front End uses the override options, as well as the user options, to generate the input streams for the other components. The local keywords in an input stream can override some of the values in the POF and ISPF profile. For example, when foreground processing is selected, the Analysis component’s ALUIN input stream is generated when you select Create Input from the Analysis JCL Processing Interface panel.

*Note*
If you do not use the Front End component to run other components, you must explicitly specify (in each component’s input stream in the JCL) any local values that you want to use that are different from the global installation option values that are defined at installation.

### Additional installation options module

When you issue the DOPTS command on the Command line of any ALTER or CHANGE MANAGER panel, the products invoke an ISPF browse session.

In this browse session, the installation option keywords and values are displayed.

*Note*
This list of installation option keywords does not include the keywords that the keywords in the POF replace.

You can copy the keywords in the browse session to a permanent data set. In your stand-alone installation options job in the installation JCL data set, specify the name of this permanent data set as the data set name in the /*ASM.SYSIN DD card.
How ALTER or CHANGE MANAGER use options

Figure 54: How ALTER or CHANGE MANAGER use options

- Installation options module
- Product options file (POF)
- User POF
- ISPf profile
- Modified by product user options
- Modified copy of ISPf profile
- Modified by component override options
- Edited in JCL or edited directly*
- Input streams

Global keywords
- Generated at installation

User keywords
- Created by user

Local or session keywords
- Generated by components (Analysis, Baseline, CM/PILOT, Compare, Import, JCL Generation, Execution)

* The only input stream that you can edit is the ALUIN input stream in foreground Analysis
Data set allocation

When you use the Front End to run components of ALTER or CHANGE MANAGER, you specify data sets for JCL, diagnostic output, worklist files, and report files.

For CHANGE MANAGER, you also specify data sets for Change Definition Language (CDL) files.

When you specify these data sets, use the guidelines in Table 3 on page 144.

Note
You must predefine partitioned data sets (PDSs). If a sequential file or member of a PDS does not exist, the product dynamically creates or allocates it.

Table 3: Guidelines for data sets

<table>
<thead>
<tr>
<th>Data set</th>
<th>Type</th>
<th>Record format</th>
<th>Record length</th>
</tr>
</thead>
<tbody>
<tr>
<td>JCL</td>
<td>Sequential file</td>
<td>Fixed length (FB)</td>
<td>80 characters</td>
</tr>
<tr>
<td></td>
<td>Member of PDS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Diagnostic output</td>
<td>Sequential file</td>
<td>Variable length (VB)</td>
<td>137 characters</td>
</tr>
<tr>
<td>Worklist</td>
<td>Sequential file</td>
<td>Fixed length (FB)</td>
<td>80 characters</td>
</tr>
<tr>
<td></td>
<td>Member of PDS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CDL file</td>
<td>Sequential file</td>
<td>Fixed length (FB)</td>
<td>80 characters</td>
</tr>
<tr>
<td></td>
<td>Member of PDS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Report</td>
<td>Sequential file</td>
<td>Fixed length (FBA)</td>
<td>80 characters</td>
</tr>
<tr>
<td></td>
<td>Member of PDS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Object names

You must name an ALTER or CHANGE MANAGER object when you create it.
Work IDs, profiles, and baselines have a two-part name in the following format:

owner.identifier

The owner can be from one to eight characters, and the identifier can be from one to 18 characters. The two parts are subject to the same restrictions as DB2 identifiers: you cannot use delimited identifiers or double-byte character set (DBCS) characters. For information about delimited identifiers, see “Delimited identifiers” on page 147. If you do not explicitly specify the owner, the owner defaults to your user ID.

**Note**
The implicit owner for a baseline is the owner of the profile that was used to create the baseline.

### Wildcard patterns

Wildcard patterns are used throughout the product to specify a partial match in the name of an object.

Table 4 on page 145 describes how wildcard patterns can be used in various components of ALTER and CHANGE MANAGER.

**Table 4: Using wildcards**

<table>
<thead>
<tr>
<th>Component</th>
<th>Use of wildcards</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specification</td>
<td>To produce lists of objects</td>
</tr>
<tr>
<td>Scope rules</td>
<td>In object names, to include lists of objects under a single scope rule</td>
</tr>
<tr>
<td>Change rules</td>
<td>To find object names that changes are applied to and to supply old and new change rule values</td>
</tr>
</tbody>
</table>

Table 5 on page 145 lists the wildcard characters that are supported for the Specification filters.

**Table 5: Wildcard characters for Specification filters**

<table>
<thead>
<tr>
<th>Wildcard character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>? (question mark)</td>
<td>Matches any single character, if the data type of the object is fixed length or variable length. For example, a value of AB?D or AB_D matches ABCD or AB1D, but not ABD.</td>
</tr>
</tbody>
</table>
Wildcard character | Description
--- | ---
% (percent sign) | Matches any string of zero or more characters, depending on the data type of the object. If the data type of the object is fixed length, an embedded wildcard matches any single character. For example, a value of %DB or *DB for a database name matches ADB or BDB, but not DB or XYZDB. A trailing wildcard matches a string of zero or more characters. For example, a value of %DB* for a database name matches ADB or ADB123, but does not match DB123.
* (asterisk) | If the data type of the object is variable length, an embedded or a trailing wildcard matches a string of zero or more characters. For example, a value of *TB or %TB matches TB, ATB, or ABTB, but not ABTBZ.

Table 6 on page 146 lists the wildcard characters that are supported for scope rules.

Table 6: Wildcard characters for scope rules

| Wildcard character | Description |
--- | ---|
? (question mark) | Matches any single character.
_ (underscore) | For example, a value of AB?D or AB_D matches ABCD, AB1D, but not ABD. |
% (percent sign) | Matches any string of zero or more characters.
* (asterisk) | For example, a value of AB*D or AB%D matches ABCD, AB123D, or ABD, but not AB. |

If you use change rules and if the new value contains wildcard characters, the current value must have exactly the same number of wildcard characters in the same order. If the new value does not contain wildcards, the current value can use any combination. Table 7 on page 146 lists the wildcard characters that are supported for change rules.

Table 7: Wildcard characters for change rules

| Wildcard character | Description |
--- | ---|
? (question mark) | Matches any single character.
_ (underscore) | For example, a current value of AB?D or AB_D matches ABCD, AB1D, but not ABD. You can create an underscore (_) as a literal instead of a wildcard by preceding it with a back slash (\). |
% (percent sign) | Matches any string of zero or more characters.
* (asterisk) | For example, a current value of AB*D or AB%D matches ABCD, AB123D, or ABD, but not AB. |

Table 8 on page 147 shows some examples of valid and invalid wildcard uses.
Table 8: Examples of valid and invalid wildcard use for change rules

<table>
<thead>
<tr>
<th>Current value</th>
<th>New value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>AB*DE</td>
<td>X*ABC</td>
<td>Valid; wildcards match</td>
</tr>
<tr>
<td>Note: Any characters found in the current value between the B and D characters are inserted between the X and A characters of the new value.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AB*DE</td>
<td>X<em>A</em></td>
<td>Invalid; number of wildcards do not match</td>
</tr>
<tr>
<td>AB<em>DE</em></td>
<td>X*ABC</td>
<td>Invalid; number of wildcards do not match</td>
</tr>
<tr>
<td>AB*DE</td>
<td>ABPROD</td>
<td>Valid; no wildcards in new value</td>
</tr>
<tr>
<td>DBTEST</td>
<td>AB*DE</td>
<td>Invalid; no wildcards in original value</td>
</tr>
<tr>
<td>A*B_</td>
<td>X_Y*</td>
<td>Invalid; wildcards out of order</td>
</tr>
</tbody>
</table>

**Delimited identifiers**

ALTER or CHANGE MANAGER support the use of delimited identifiers for DB2 objects, but not for BMC objects, such as work IDs, baselines, baseline profiles, and migrate profiles.

The double quotation mark (") is the only escape character that is supported.

Delimited identifiers can also be used in data definition language (DDL), CDL, worklists, Data Manipulation Language (DML), and for reserved words. ALTER and CHANGE MANAGER evaluate the identifiers to determine whether delimiters are required. If delimiters are not required, they are removed.

Delimited identifiers are supported to specify owners of DB2 objects and to specify the names of certain types of DB2 objects. You can use delimited identifiers to specify DB2 object names that use long identifiers. These objects include tables, table columns, indexes, aliases, synonyms, views, view columns, triggers, and trigger columns. Names for databases and table spaces cannot be named using delimited identifiers. Owners of these DB2 objects, however, can be named using delimited identifiers. The following example illustrates the use of a delimited identifier:

```
000008 -SQL 000450 CREATE TABLE NCH.TBLARGE
000009                      ("SYNONYM"  CHAR(15) NOT NULL,
```

**Note**

Twenty bytes are allowed for object names and ten bytes for owner names specified with delimited identifiers; two bytes are used for the escape characters. If you use delimited identifiers to specify names, embedded double quotation marks within a name might cause truncation. Two consecutive double quotation marks are required to represent one double quotation mark within a delimited identifier.
Wildcard characters that are specified within delimiters are considered wildcard characters, not literal characters. The asterisk (*) specified for a STOGROUP volume ID is supported for users of the IBM Storage Management Subsystem (SMS).

For additional information about using delimited identifiers, see the IBM documentation.

Nonprintable or nonviewable characters

You must specify nonprintable and nonviewable characters (such as null and control characters) in limit keys, view text, trigger text, or check constraint text in an external hexadecimal format.

These characters are not indicated in literal strings. For example, the EBCDIC string '2 1' does not indicate that the second character is null; the string appears to be blank. The string should be specified as X'F200F1'.

Table 9 on page 148 provides the hexadecimal formats for common nonprintable or nonviewable characters.

<table>
<thead>
<tr>
<th>Character</th>
<th>Hexadecimal format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null</td>
<td>X'00'</td>
</tr>
<tr>
<td>Horizontal tab ('HT')</td>
<td>X'05'</td>
</tr>
<tr>
<td>Form feed</td>
<td>X'0C'</td>
</tr>
<tr>
<td>Carriage return</td>
<td>X'0D'</td>
</tr>
<tr>
<td>New line ('NL')</td>
<td>X'15'</td>
</tr>
<tr>
<td>Line feed ('LF')</td>
<td>X'25'</td>
</tr>
</tbody>
</table>

Note the following items about nonviewable characters:

- If a value in a limit key field in a partitioned index contains a nonviewable character, you cannot edit that value in the Specification component.
- If null characters exist in the internal format of a value for an object, the Baseline, Import, and Compare components issue a warning message.
Access to components

Maintaining security over the components is an important consideration.

Executing a worklist can change the definitions of your DB2 objects as they are defined in the DB2 system catalog tables. To control access to the components, you control the authorization that is granted to plans for the components.

Plans that are included with ALTER and CHANGE MANAGER provide access to the Front End, Import, Specification, Analysis, Baseline, Compare, and Execution components. You can control whether users can access ALTER and CHANGE MANAGER by controlling the authorization granted to each of these plans.

ALTER and CHANGE MANAGER plan names have the format prdvryz, where prd is the product code, vr is the version and release, y is the access type, and z is the component.

The Execution component plan names have the format prdvrmn, where prd is the product code, vr is the version and release, and mn is a unique plan identifier.

ALTER and CHANGE MANAGER plans

ALTER and CHANGE MANAGER use plans to provide access to the Front End, Import, Specification, and Analysis components.

Table 10 on page 149 lists the plans that the components of ALTER and CHANGE MANAGER use.

Table 10: ALTER and CHANGE MANAGER plans

<table>
<thead>
<tr>
<th>Plan name</th>
<th>Component name</th>
<th>Plan description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACMvryA</td>
<td>Analysis</td>
<td>Enables users to access the Analysis component to analyze changes and generate worklists. Users cannot make changes to DB2 objects and data by creating a worklist. However, the Specification and Analysis plans provide a way for users to request and analyze changes. For this reason, administrators can usually place minimum restrictions on using Specification and Analysis.</td>
</tr>
<tr>
<td>ACMvryE</td>
<td>Environment</td>
<td>Enables users to use the ENVI command to review the system environment. Because this plan is not accessed outside of the other components, administrators can usually grant PUBLIC access to this plan. This plan does not control the use of the ENV keyword with batch components.</td>
</tr>
</tbody>
</table>
### Plan name | Component name | Plan description
--- | --- | ---
ACMvryF | Front End | Controls access to the ISPF interface
Authority to use this plan enables users to create and maintain the product's objects and perform other Front End functions. If the Distributed Data Facility (DDF) is installed, the -BIND statement for this plan adds a reference to the remote package list.

ACMvryI | Import | Enables users to import files into the Change Definition (CD) tables
This plan accesses the Import component. For ALTER, the plan enables users to import data definition language (DDL) into the CD tables. For CHANGE MANAGER, it enables users to import DDL, Change Definition Language (CDL), and worklists into the CD tables.

ACMvryS | Specification | Enables users to request changes to database objects and data
This plan accesses the Specification component. With access to Specification, users can also search and review groups of objects.
Note: While Specification does not enable users to actually perform changes, it does enable them to specify changes and to view existing data structures. If the Distributed Data Facility (DDF) is installed, the -BIND statement for this plan adds a reference to the remote package list.

### CHANGE MANAGER plans

CHANGE MANAGER uses additional plans to provide access to the Baseline, Compare, Baseline Report, and LOB DATA MOVER components.

Table 11 on page 150 lists additional plans that the components of CHANGE MANAGER use.

#### Table 11: CHANGE MANAGER plans

<table>
<thead>
<tr>
<th>Plan name</th>
<th>Component name</th>
<th>Plan description</th>
</tr>
</thead>
</table>
| ACMvryB | Baseline | Enables users to establish and delete baselines
Because baselines (especially full-recovery baselines) contain critical information that is used for database recovery, administrators should allow only informed users access to this plan. |
<table>
<thead>
<tr>
<th>Plan name</th>
<th>Component name</th>
<th>Plan description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACMvryC</td>
<td>Compare</td>
<td>Enables users to compare data structure definitions Because no DB2 data structures are modified, administrators can usually grant unlimited access</td>
</tr>
<tr>
<td></td>
<td></td>
<td>to this plan. If the DDF is installed, the -BIND statement for this plan adds a reference to the remote package list.</td>
</tr>
<tr>
<td>ACMvryR</td>
<td>Baseline Report</td>
<td>Enables users to create baseline reports Grant unlimited access to this plan.</td>
</tr>
<tr>
<td>ACMvryL</td>
<td>LOB DATA MOVER</td>
<td>When the Database Administration solution is installed, enables the LOB DATA MOVER program to attach to DB2 The LOB DATA MOVER program unloads and loads data that is contained in large object (LOB) columns.</td>
</tr>
</tbody>
</table>

### Execution component plans

You can restrict access to functions of the Execution component by using PLAN authorizations.

Table 12 on page 151 lists the plans that the Execution component uses.

#### Table 12: Execution plans

<table>
<thead>
<tr>
<th>Plan name</th>
<th>Function name</th>
<th>Plan description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AEXvHA (ALTER and CHANGE MANAGER)</td>
<td>Execution Monitor Entry (Authorization)</td>
<td>Enables users to execute a worklist when EXECUTE authority is granted You should carefully consider who receives authorization to use this plan.</td>
</tr>
<tr>
<td>Plan name</td>
<td>Function name</td>
<td>Plan description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| AEXvrHM (ALTER and CHANGE MANAGER) | Execution Monitor | Enables users to attach to DB2 with alternate authorization IDs for the -AUTH commands. This plan does not control who has authorization to execute a worklist. Because this plan does not affect who can run Execution, you can grant PUBLIC authority to this plan. The Execution plan contains some packages that use dynamic SQL. Some of these packages cause long-running SQL and might need to be added to your resource limit specification table (RLST). The packages are as follows:  
  - AEXAUNLD unloads data from tables.  
  - AEXSQLIO performs all worklist -SQL commands, including deletions before a -LOAD or -BMCL command that migrates data only.  
  - AEXESTDL performs some of the restart logic before restarting a -LOAD command, including deleting previously loaded rows.  
By restricting authorization to run the Execution plans, you can control what change and migrate functions users can perform. For example, by granting unlimited access to Specification and Analysis while controlling access to the Execution Monitor Entry, you can allow your users to run ALTER for training purposes or use it as a system dictionary. The Execution Security Exit provides further control over the Execution component’s authorization switching function. |

**Authorization switching**

The Execution component performs an authorization switching function to ensure that DB2 structures are created with the proper owner and creator.
To secure this function, you can perform one of the following activities:

- Restrict the EXECUTE authority on the Execution Monitor Entry plan. This restriction will prevent unauthorized users from being able to run Execution. For the highest level of security, restrict access to the Execution Monitor Entry plan.

- Create an Execution security exit to add site-specific restrictions to authorization switching.

Execution calls the security exit at various points to allow installation security checking and option enforcement.

Typical uses of this exit include:

- Enforcing hash checking so that users are not allowed to modify worklists
- Switching the authorization ID of the execution run user at initialization
- Preventing or restricting user authorization ID switching
- Preventing user modification of SQL GRANT commands
- Inspecting and modifying SQL commands before they are executed

The assembler source for the security exit is distributed in the DBSAMP library (member ALUEUSX1). To enable the security exit, modify ALUEUSX1, then assemble and link the member. This module is loaded during initialization.

ALUEUSX1 communicates via Registers 0, 1, and 15. When ALUEUSX1 is called, Register 0 contains a function code. Register 1 contains the address of a user exit control block that AEXUSERX DSECT (in the sample program) describes. ALUEUSX1 returns a value in Register 15 that informs Execution how to proceed.

For more information about switching authorization, see the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.

**Execution security exit function codes**

The table in this topic lists and describes the Register 0 function codes.

The sample exit includes detailed documentation about register usage, DSECT mapping, and call conventions. For a list of return codes, see the messages library in the BMC Documentation Center (http://www.bmc.com/available/documentation-center.html).
<table>
<thead>
<tr>
<th>Code</th>
<th>Call</th>
<th>Occurs</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'01'</td>
<td>Initialization</td>
<td>During Execution initialization</td>
<td>At this point, you can inspect the user ID that is running the job, change it if necessary, or halt processing. You can also enable hash checking to prevent a user from making any modifications to a worklist (other than deleting entire commands) throughout the execution run. If you need storage for local routines, you should perform a GETMAIN at this time. Store the address in the user exit control block AEXUSERX, as described in the code sample.</td>
</tr>
<tr>
<td>X'02'</td>
<td>-AUTH command switch</td>
<td>Immediately before an authorization ID switch</td>
<td>The authorization ID might be inspected or changed, or processing might be halted. If the AEXEFAUH bit of flag ASXEFLAG in the user exit DSECT is set, hash checking has failed on the current -AUTH command. If the return code from this exit is an odd number (that is, the low order 1 bit is set), the command that executed under the switch is verified to ensure that the hash code is correct. This verification prevents you from changing commands that are executed under a switch. If this bit is off (return code 0 or 4), hash checking is done as specified for the main run. If this bit is on (return code 1 or 5), then hash checking is forced for the duration of the switch. In this case, Execution halts when it encounters a modified or new statement in the worklist. If the return code low order 2 bit is on, the command that is executed under the switch is allowed to fail hash checking provided that the next command is not an SQL GRANT. This restriction means that you can modify the worklist as needed but cannot modify any GRANT statements. The low order 2 bit in the return code indicates whether to enforce hash failures when an SQL GRANT command is processed. Hash checking is enforced for the duration of the switch. In this case, Execution halts when it encounters a modified or new SQL GRANT statement in the worklist.</td>
</tr>
<tr>
<td>X'03'</td>
<td>termination/cleanup</td>
<td>Immediately before termination</td>
<td>The call enables you to perform any necessary cleanup functions. For example, if you used another security product to perform security validations, you might want to sign off. Remember to perform a FREEMAIN to free any storage that you allocated. Return codes are ignored.</td>
</tr>
<tr>
<td>Code</td>
<td>Call</td>
<td>Occurs</td>
<td>Notes</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------</td>
<td>---------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| X'04' | SQL inspection        | Immediately before invoking dynamic SQL     | The call enables you to inspect the SQL command that is about to be executed. You can use the command as it is, replace it with another command, or stop processing.  
If the AEXEFAUH bit of flag ASXFLAG in the user exit DSECT is set, hash checking has failed on the -SQL command.  
The SQL call passes the address of the SQL statement in field ASXSQL. This address points to a field of the format $hhx...x$, where $hh$ is a halfword length field, and $x...x$ is the SQL statement about to be executed. This field is passed to DB2 for execution. An SQL failure halts execution. |
| X'05' | -SETS command inspection | Immediately before executing a -SETS (SET CURRENT SQLID) worklist command | You can inspect, change, or deny the SQLID that is about to be switched.  
If the AEXEFAUH bit of flag ASXFLAG in the user exit DSECT is set, hash checking has failed on the -AUTH command.  
This call does not affect hash checking because the SET CURRENT SQLID mechanism is inherently secured by the DB2 system itself. The primary purpose of this call is to allow the use of special SQLIDs that are not the creator or owner of DB2 objects that are being created in the worklist. |
| X'06' | -GLID command inspection | Immediately before executing a -GLID (set global authorization ID) worklist command | You can inspect, change, or deny the global authorization ID (GLID) that is about to be switched.  
In addition, you can activate hash checking and other options on this exit call.  
If the AEXEFAUH bit of flag ASXFLAG in the user exit DSECT is set, hash checking has failed on the -AUTH command. |

**Execution security exit testing**

Before you enable your security exit, you should test the exit.

Following are some suggestions to help you develop and test your security exit:

- Test the exit initially with simple logic and test cases to verify that you are correctly processing the function codes and return codes.
- Add the DEBUG keyword to the AEXIN parameter input.  
Message BMC16990I is issued to AEXPRINT and includes the following information:
— Function code with which the exit is called
— WORKID or SQLID that is about to be switched
— USERID that is associated with the job itself
— Return code (Register 15) that is returned from the exit
— WORKID or SQLID that is returned from the exit

Contact BMC Customer Support if you have questions about the Execution security exit.

Access to the worklist parallelism feature

With the Database Administration solution, you can use the worklist parallelism feature to execute portions of a worklist concurrently.

CHANGE MANAGER uses the BMC Cross-System Image Manager (XIM) technology to provide sysplex performance improvements by enabling the distribution and management of discrete units of work (UOW) across one or more IBM OS/390 and z/OS images.

By default, user access to execute portions of a worklist concurrently and to dynamically start XIM is not restricted. You can control access to these functions for a user or a group of users by performing the following tasks:

1. Apply a zap.
2. If you are using the IBM Resource Access Control Facility (RACF), specify a general resource profile.

Note
If you are using another security package that is compatible with the System Authorization Facility (SAF), contact BMC Customer Support.

Zap to restrict access to worklist parallelism

To enable the restriction of access to worklist parallelism functions, you must apply a zap.
Apply the following zap to the Execution function of CHANGE MANAGER:

<table>
<thead>
<tr>
<th>NAME</th>
<th>AEXPMAIN MAINRACC</th>
</tr>
</thead>
<tbody>
<tr>
<td>VER</td>
<td>003E 47F0, C1D8</td>
</tr>
<tr>
<td>REP</td>
<td>003E 4700, 0000</td>
</tr>
<tr>
<td>CHECKSUM</td>
<td>0916482E</td>
</tr>
</tbody>
</table>

**General resource profile for worklist parallelism**

In RACF, general resource profiles are used to protect the resources that are defined in the class descriptor table, such as programs.

To restrict a user’s or group’s access to each of the worklist parallelism functions, you must add general resource profiles with the following profile information:

- CLASS => FACILITY
- PROFILE => BMCACM.ssid.PARALLEL.object

The profile definition contains the following values:

- BMCACM specifies that the profile is for CHANGE MANAGER.
- ssid represents the name of the DB2 subsystem or a DB2 group attachment name (wildcard characters can be used to match one or more characters).
- PARALLEL represents the function that is secured.
- object represents the object or resource name that is secured.
  - For executing a worklist, the object is EXECUTE.
  - For starting XIM dynamically, the object is DYNSTART.

Each user or group that is given access to a resource profile must have an access level of CONTROL or higher.

**Setting the MEMLIMIT system parameter**

1. Use any of the following methods to override the default value of the MEMLIMIT value:
   - Specify the MEMLIMIT parameter in the JCL
- Specify REGION=0M in the JCL
- Use the SMF IEFUSI exit

**Note**
The default value for the System Management Facility (SMF) MEMLIMIT parameter is 2 GB. This value is set in member SMFPRMxx in SYS1.PARMLIB.

---

**Modification of ISPF skeletons and variables**

The JCL Generation component generates JCL by using a method that improves upon standard ISPF file-tailoring services. The method used is based on ISPF skeletons.

The $AJXDOC member in the HLQ.BMCSLIB data set lists and briefly describes each non-DD and DD statement skeletons that JCL Generation uses. The member also lists the variables that JCL Generation uses to construct the default names for permanent data sets.

Each skeleton name in the $AJXDOC member has a corresponding member in the HLQ.BMCSLIB data set. Some of the members that contain skeletons also provide comments, tips, and suggestions for using the skeletons.

You can specify up to five user-defined variables in the POF. The JCL Generation User Defined Variables panel allows you to specify variable names (up to eight characters). Each variable has a corresponding symbolic variable, as shown in Table 14 on page 158. You can use the symbolic variables in your job cards or data set prefixes. For information about specifying the variables, see “Setting user variables” on page 216.

**Table 14: User-defined variables**

<table>
<thead>
<tr>
<th>POF variable</th>
<th>Symbolic variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER_VAR1_CHAR</td>
<td>&amp;UVR1</td>
</tr>
<tr>
<td>USER_VAR2_CHAR</td>
<td>&amp;UVR2</td>
</tr>
<tr>
<td>USER_VAR3_CHAR</td>
<td>&amp;UVR3</td>
</tr>
<tr>
<td>USER_VAR4_CHAR</td>
<td>&amp;UVR4</td>
</tr>
<tr>
<td>USER_VAR5_CHAR</td>
<td>&amp;UVR5</td>
</tr>
</tbody>
</table>

The *ALTER and CHANGE MANAGER for DB2 Reference Manual* describes the symbolic and SLIB variables that JCL Generation uses and specifies the length of each variable.
To improve the performance of the JCL construction phase of JCL Generation, BMC uses an SLIB compiler. Consequently, *if you edit SLIBs after installation, you must recompile them*. For information, see *ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2*.

## JCL Generation data sets sizing function

You can use the JCL Generation data set sizing function to tailor the data set sizes when the JCL is built.

The function gathers information from one of the following sources (shown in general order of accuracy, from most accurate to least accurate):

1. DASD MANAGER PLUS product statistics database (statistics that the BMCSTATS utility gathers)
2. DB2 system catalog (statistics that the IBM RUNSTATS utility gathers)
3. Results of VSAM object sampling
4. Default data set allocation parameters that are set from the JCL Generation Individual Data Set Options panel

Whether it uses the BMCSTATS historical database or object sampling, or even if it does not use sizing, JCL Generation obtains some information from the DB2 system catalog.

The formulas for estimating data set size are taken from the documentation for the IBM DB2 utilities and from the documentation for the BMC utilities. Table 15 on page 159 shows the statistics that JCL Generation uses for space estimation and the source of the statistics.

### Table 15: Data set sizing values and sources

<table>
<thead>
<tr>
<th>Value</th>
<th>BMCSTATS</th>
<th>IBM RUNSTATS</th>
<th>VSAM sampling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of active pages</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Number of modified pages</td>
<td>X</td>
<td>NA</td>
<td>X</td>
</tr>
<tr>
<td>Page size</td>
<td>NA</td>
<td>X</td>
<td>NA</td>
</tr>
<tr>
<td>Maximum row length</td>
<td>NA</td>
<td>X</td>
<td>NA</td>
</tr>
<tr>
<td>Average row length</td>
<td>X</td>
<td>NA</td>
<td>X</td>
</tr>
</tbody>
</table>
### Product table maintenance

You are responsible for maintaining the tables that are used by ALTER or CHANGE MANAGER.

In addition to installing the tables, maintenance tasks for the tables include:

- Sizing the tables to accommodate growth requirements
- Purging the tables of old or unneeded data
- Updating the rename table
- Reorganizing the tables to improve performance
- Backing up for recovery

For more information about the administrative tasks associated with managing the application databases, see the IBM documentation.

CHANGE MANAGER uses the tables that are listed in Table 17 on page 161. ALTER uses the all of the CD tables and some of the CM tables. For versions 10.1 and later, these tables store the version and release of the installed product. Earlier versions prefixed the table names with V vr, where the vr reflected the product version and release.

The products use the prefixes listed in the default table name (Table 16 on page 161).
### Table 16: Table prefixes

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>BL</td>
<td>Baseline component of CHANGE MANAGER</td>
</tr>
<tr>
<td>CD</td>
<td>Specification and Analysis components of ALTER and CHANGE MANAGER</td>
</tr>
<tr>
<td>CM</td>
<td>Objects for ALTER and CHANGE MANAGER</td>
</tr>
<tr>
<td>CP</td>
<td>CM/PILOT component of CHANGE MANAGER</td>
</tr>
</tbody>
</table>

### Table 17: ALTER and CHANGE MANAGER tables

<table>
<thead>
<tr>
<th>Default table name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BL_ALIAS</td>
<td>Alias definitions</td>
</tr>
<tr>
<td>BL_AUXTABLE</td>
<td>Auxiliary object (such as LOB table space, auxiliary table, and auxiliary index) definitions</td>
</tr>
<tr>
<td>BL_CHECK</td>
<td>Check constraint definitions</td>
</tr>
<tr>
<td>BL_CHECK2</td>
<td>Path schema for check constraints</td>
</tr>
<tr>
<td>BL_COLAUTH</td>
<td>Column authorizations</td>
</tr>
<tr>
<td>BL_COLUMN</td>
<td>Table and view column definitions</td>
</tr>
<tr>
<td>BL_DATABASE</td>
<td>Database definitions</td>
</tr>
<tr>
<td>BL_DBAUTH</td>
<td>Security information for databases in a baseline</td>
</tr>
<tr>
<td>BL_INDEX</td>
<td>Index definitions</td>
</tr>
<tr>
<td>BL_KEY</td>
<td>Index key definitions</td>
</tr>
<tr>
<td>BL_KEYTARGETS</td>
<td>Key-targets for extended index definitions</td>
</tr>
<tr>
<td>BL_PACKAGE</td>
<td>Package definitions</td>
</tr>
<tr>
<td>BL_PARMS</td>
<td>Stored procedure parameter definitions</td>
</tr>
<tr>
<td>BL_REL</td>
<td>Foreign key definitions</td>
</tr>
<tr>
<td>BL_RELKEY</td>
<td>Foreign key column definitions</td>
</tr>
<tr>
<td>BL_RESAUTH</td>
<td>Security information for table spaces and storage groups in this baseline</td>
</tr>
<tr>
<td>BL_ROUTINES</td>
<td><em>(ALTER and CHANGE MANAGER version 11.1 and later)</em> Stored procedure definitions</td>
</tr>
<tr>
<td>BL_ROUTINESTEXT</td>
<td>LOB table for baseline stored procedure text</td>
</tr>
<tr>
<td>BL_RTAUTH</td>
<td>Privileges granted on stored procedures in a baseline</td>
</tr>
<tr>
<td>BL_SEQUENCES</td>
<td>Identity column definitions</td>
</tr>
<tr>
<td>BL_SQAUTH</td>
<td>Privileges granted on sequence objects in this baseline</td>
</tr>
<tr>
<td>Default table name</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>BL_STOGROUP</td>
<td>Storage group definitions</td>
</tr>
<tr>
<td>BL_SYNONYM</td>
<td>Synonym definitions</td>
</tr>
<tr>
<td>BL_TABAUTH</td>
<td>Security information for tables and views in this baseline</td>
</tr>
<tr>
<td>BL_TABLE</td>
<td>Table definitions</td>
</tr>
<tr>
<td>BL_TABLEPART</td>
<td>Table and index space partition definitions</td>
</tr>
<tr>
<td>BL_TABLESPACE</td>
<td>Table space definitions</td>
</tr>
<tr>
<td>BL_TRIG_COL</td>
<td>Trigger column definitions</td>
</tr>
<tr>
<td>BL_TRIG_TEXT</td>
<td>Trigger text definitions</td>
</tr>
<tr>
<td>BL_TRIG_TEXT_AUX</td>
<td>Trigger text definitions for auxiliary tables</td>
</tr>
<tr>
<td>BL_TRIGGER</td>
<td>Trigger definitions</td>
</tr>
<tr>
<td>BL_UNQCNST</td>
<td>Unique constraint definitions</td>
</tr>
<tr>
<td>BL_UNQCNST_COL</td>
<td>Unique constraint column definitions</td>
</tr>
<tr>
<td>BL_VIEW</td>
<td>View definitions</td>
</tr>
<tr>
<td>BL_VIEW_TEXT</td>
<td>View text definitions</td>
</tr>
<tr>
<td>BL_VIEW_TEXT_AUX</td>
<td>View text definitions for auxiliary tables</td>
</tr>
<tr>
<td>BL_VOLS</td>
<td>Additional volume information for storage groups and table partitions</td>
</tr>
<tr>
<td>CD_ALIAS</td>
<td>Alias definitions</td>
</tr>
<tr>
<td>CD_AUXTABLE</td>
<td>Auxiliary object (such as LOB table space, auxiliary table, and auxiliary index) definitions</td>
</tr>
<tr>
<td>CD_CHECK</td>
<td>Check constraint definitions</td>
</tr>
<tr>
<td>CD_CHECK2</td>
<td>Path schema for check constraints</td>
</tr>
<tr>
<td>CD_COLUMN</td>
<td>Table and view column definitions</td>
</tr>
<tr>
<td>CD_DATABASE</td>
<td>Database definitions</td>
</tr>
<tr>
<td>CD_INDEX</td>
<td>Index definitions</td>
</tr>
<tr>
<td>CD_KEY</td>
<td>Index key definitions</td>
</tr>
<tr>
<td>CD_KEYTARGETS</td>
<td>Key-targets for extended index definitions</td>
</tr>
<tr>
<td>CD_PACKAGE</td>
<td>Package definitions</td>
</tr>
<tr>
<td>CD_PARMS</td>
<td>Stored procedure parameter definitions</td>
</tr>
<tr>
<td>CD_REL</td>
<td>Foreign key definitions</td>
</tr>
<tr>
<td>CD_RELKEY</td>
<td>Foreign key column definitions</td>
</tr>
<tr>
<td>CD_ROUTINES</td>
<td>Stored procedure definitions</td>
</tr>
<tr>
<td>Default table name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CD_ROUTINES2</td>
<td><em>(ALTER and CHANGE MANAGER version 11.1 and later)</em> Stored procedure definitions</td>
</tr>
<tr>
<td>CD_ROUTINES2NEWTEXT</td>
<td>New text of stored procedure</td>
</tr>
<tr>
<td>CD_ROUTINES2TEXT</td>
<td>Text of stored procedure</td>
</tr>
<tr>
<td>CD_SEQUENCES</td>
<td>Identity column definitions</td>
</tr>
<tr>
<td>CD_STOGROUP</td>
<td>Storage group definitions</td>
</tr>
<tr>
<td>CD_SYNONYM</td>
<td>Synonym definitions</td>
</tr>
<tr>
<td>CD_TABLE</td>
<td>Table definitions</td>
</tr>
<tr>
<td>CD_TABLEPART</td>
<td>Table and index space partition definitions</td>
</tr>
<tr>
<td>CD_TABLESPACE</td>
<td>Table space definitions</td>
</tr>
<tr>
<td>CD_TRIG_COL</td>
<td>Trigger column definitions</td>
</tr>
<tr>
<td>CD_TRIG_TEXT</td>
<td>Trigger text definitions</td>
</tr>
<tr>
<td>CD_TRIG_TEXT_AUX</td>
<td>Trigger text definitions for auxiliary tables</td>
</tr>
<tr>
<td>CD_TRIGGER</td>
<td>Trigger definitions</td>
</tr>
<tr>
<td>CD_UNQCNST</td>
<td>Unique constraint definitions</td>
</tr>
<tr>
<td>CD_UNQCNST_COL</td>
<td>Unique constraint column definitions</td>
</tr>
<tr>
<td>CD_VIEW</td>
<td>View definitions</td>
</tr>
<tr>
<td>CD_VIEW_TEXT</td>
<td>View text definitions</td>
</tr>
<tr>
<td>CD_VIEW_TEXT_AUX</td>
<td>View text definitions for auxiliary tables</td>
</tr>
<tr>
<td>CD_VOLS</td>
<td>Additional volume information for storage groups and table parts</td>
</tr>
<tr>
<td>CM_BASELINE</td>
<td>Baseline header information</td>
</tr>
<tr>
<td>CM_BLPROFILE</td>
<td>Baseline profile information</td>
</tr>
<tr>
<td>CM_HISTORY</td>
<td>Comparison statistical information</td>
</tr>
<tr>
<td>CM_LOCATION</td>
<td>Location information for migrate profiles</td>
</tr>
<tr>
<td>CM_MIGPROFILE</td>
<td>Migrate profile information</td>
</tr>
<tr>
<td>CM_RENAME</td>
<td>Old and new names of renamed objects</td>
</tr>
<tr>
<td>CM_RULES</td>
<td>Rules for migrate profiles</td>
</tr>
<tr>
<td>CM_SCOPE</td>
<td>Scope for migrate and baseline profiles</td>
</tr>
<tr>
<td>CM_SECURITY</td>
<td>Security table</td>
</tr>
<tr>
<td>CM_SYNC</td>
<td>Execution sync table</td>
</tr>
<tr>
<td>CM_UNLOADDSN</td>
<td>Information about unloaded data sets used in a full-recovery baseline</td>
</tr>
</tbody>
</table>
Table size maintenance

The baseline tables (those named BL_ name) of CATALOG MANAGER require more ongoing maintenance than other tables.

Because the baseline tables hold critical information used for database recovery, information in them should be retained as long as the baselines are needed. If your installation regularly establishes baselines, these tables grow over time. You must ensure that these tables are reorganized and that the size of the tables is expanded on a regular basis. To help control the baseline table, you can delete any unwanted baselines. For more information about deleting baselines, see the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.

The sync tables of ALTER and CATALOG MANAGER (CM_SYNC) and the CM/PILOT component (CP_SYNC) also require more ongoing maintenance. The records in the sync table for a particular work ID or task ID are used as progress indicators during execution of a worklist. These tables grow over time. You will need to ensure that these tables are reorganized and expanded on a regular basis. You will also need to delete any work IDs and task IDs that are no longer needed.

**Tip**

You can also use member ALUDSYNC in the HLQUBMCCNTL data set to maintain the CM_SYNC and CP_SYNC tables.

To help control the size of the sync tables, you can delete any unwanted work IDs or task IDs. When you delete a work ID, all of the change and migration requests for that work ID are deleted from the product's CD tables. If the status of the work ID is Exec Strt (for Execution started) or Exec Comp (for Execution complete), all of the rows that are written to the product's work ID tables (such as the sync table) are also

<table>
<thead>
<tr>
<th>Default table name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CM_WORKID</td>
<td>Information about each work ID</td>
</tr>
<tr>
<td>CM_XML</td>
<td>Row of XML data for object definitions</td>
</tr>
<tr>
<td>CM_XML_AUX</td>
<td>XML data definitions for auxiliary tables</td>
</tr>
<tr>
<td>CP_APPLICATION</td>
<td>Information about each CM/PILOT application</td>
</tr>
<tr>
<td>CP_DML_STMT</td>
<td>Information about each DML statement in each CM/PILOT task ID</td>
</tr>
<tr>
<td>CP_SCRIPT</td>
<td>Information about user-defined CM/PILOT scripts</td>
</tr>
<tr>
<td>CP_STEP</td>
<td>Information about steps in user-defined CM/PILOT scripts</td>
</tr>
<tr>
<td>CP_SYNC</td>
<td>Sync table for CM/PILOT worklist execution</td>
</tr>
<tr>
<td>CP_TASKID</td>
<td>Information about each CM/PILOT task ID</td>
</tr>
</tbody>
</table>
deleted. Deleting the work ID from the product tables does not affect committed changes that were made to the DB2 catalog during execution of the work ID. (For information about viewing the status of the worklist, see the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.)

---

**Note**
Deleting a work ID with Exec Strt status is not recommended because the work ID cannot be restarted for completion.

---

**WARNING**
Full-recovery baselines are associated with alter-type work IDs. If you delete a work ID that has an associated full-recovery baseline, the work ID for that baseline will become null.

---

**Tip**
To delete work IDs or task IDs, see the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.

---

**Table security**

Only those users who need to perform diagnostic functions on ALTER or CHANGE MANAGER should be given access to the ALTER or CHANGE MANAGER database objects.

Access to the database through ALTER and CHANGE MANAGER functions is achieved through EXECUTE authority on plans, except on the rename table. The Compare component uses the CHANGE MANAGER rename table (CM_RENAME) when a baseline is used as input to resolve the names of renamed objects. If users change the names of DB2 objects outside of CHANGE MANAGER, they should update the CHANGE MANAGER rename table so that the Compare component can correctly resolve renamed objects. To update the table, the users must submit SQL INSERT statements through a DB2 access tool, and must therefore have authority to update the rename table. Table 18 on page 166 shows the SQL INSERT statements for manually updating the CHANGE MANAGER rename table. Access to the rename table is available by granting DB2 authorization to INSERT, UPDATE, DELETE and SELECT that table. For more information about renamed object resolution, see the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.

---

**WARNING**
Failure to properly update the rename table could result in data loss, because Compare generates DROP and CREATE statements rather than an ALTER statement for renamed objects.
### Table 18: SQL for updating rename table

<table>
<thead>
<tr>
<th>DB2 object</th>
<th>INSERT statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage group</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NEWNAME_PART1, TIMESTMP) VALUES ('SG', 'oldName', 'newName', 'timeStamp')</td>
</tr>
<tr>
<td>Database</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NEWNAME_PART1, TIMESTMP) VALUES ('DB', 'oldName', 'newName', 'timeStamp')</td>
</tr>
<tr>
<td>Table space</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTMP) VALUES ('TS', 'oldDatabase', 'oldTableSpace', 'newDatabase', 'newTableSpace', 'timeStamp')</td>
</tr>
<tr>
<td>Table</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTMP) VALUES ('TB', 'oldCreator', 'oldName', 'newCreator', 'newName', 'timeStamp')</td>
</tr>
<tr>
<td>Table column</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, CONAME, NEWNAME_PART1, NEWNAME_PART2, NEWCONAME, TIMESTMP) VALUES ('CO', 'oldCreator', 'oldTable', 'oldColumn', 'newCreator', 'newTable', 'newColumn', 'timeStamp')</td>
</tr>
<tr>
<td>Check constraint</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NAME_PART3, NEWNAME_PART1, NEWNAME_PART2, NEWNAME_PART3, TIMESTMP) VALUES ('CK', 'oldCreator', 'oldTable', 'oldName', 'newCreator', 'newTable', 'newName', 'timeStamp')</td>
</tr>
<tr>
<td>Index</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTMP) VALUES ('IX', 'oldCreator', 'oldName', 'newCreator', 'newName', 'timeStamp')</td>
</tr>
<tr>
<td>DB2 object</td>
<td>INSERT statement</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Unique constraint</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NAME_PART3, NEWNAME_PART1, NEWNAME_PART2, NEWNAME_PART3, TIMESTMP)</td>
</tr>
<tr>
<td></td>
<td>VALUES ('UC', 'oldCreator', 'oldTable', 'oldName', 'newCreator', 'newTable', 'newName', 'newTimeStamp')</td>
</tr>
<tr>
<td>Trigger</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTMP)</td>
</tr>
<tr>
<td></td>
<td>VALUES ('TR', 'oldSchema', 'oldName', 'newSchema', 'newName', 'timeStamp')</td>
</tr>
<tr>
<td>Foreign key</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NAME_PART3, NEWNAME_PART1, NEWNAME_PART2, NEWNAME_PART3, TIMESTMP)</td>
</tr>
<tr>
<td></td>
<td>VALUES ('FK', 'oldCreator', 'oldTable', 'oldName', 'newCreator', 'newTable', 'newName', 'timeStamp')</td>
</tr>
<tr>
<td>View</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTMP)</td>
</tr>
<tr>
<td></td>
<td>VALUES ('VW', 'oldCreator', 'oldName', 'newCreator', 'newName', 'timeStamp')</td>
</tr>
<tr>
<td>View column</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, CONAME, NEWNAME_PART1, NEWNAME_PART2, NEWCONAME, TIMESTMP)</td>
</tr>
<tr>
<td></td>
<td>VALUES ('VC', 'oldCreator', 'oldView', 'oldColumn', 'newCreator', 'newView', 'newColumn', 'timeStamp')</td>
</tr>
<tr>
<td>Stored procedure</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTMP)</td>
</tr>
<tr>
<td>(external)</td>
<td>VALUES ('SP', 'oldSchema', 'oldName', 'newSchema', 'newName', 'newTimeStamp')</td>
</tr>
<tr>
<td>Stored procedure</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NAME_PART3, NEWNAME_PART1, NEWNAME_PART2, NEWNAME_PART3, TIMESTMP)</td>
</tr>
<tr>
<td>(native SQL)</td>
<td>VALUES ('SP', 'oldSchema', 'oldName', 'newSchema', 'newName', 'newTimeStamp')</td>
</tr>
</tbody>
</table>
### Setting options

You can set user and product options in ALTER and CHANGE MANAGER, CM/PILOT, and JCL Generation.

### Setting the attributes for the panel display

Use the Panel Attribute Options panel to specify your preferences for long names, entry field and error delimiters, and screen colors.

#### To set the attributes for the panel display

1. From the ALTER or CHANGE MANAGER Main Menu, select **Options** and press **Enter**.

2. From the ALTER or CHANGE MANAGER Main Options Menu panel, select **Panel Attribute Options** and press **Enter**.
The ALTER or CHANGE MANAGER Panel Attribute Options panel is displayed (Figure 55 on page 169).

**Figure 55: Panel Attribute Options panel**

```
ALUFOPT1 ----------- CHANGE MANAGER Panel Attribute Options -------------------
Command ===> Type information. Then press Enter to continue or PF12 for previous panel.
Zoom/Truncation Attributes.
  Allow popup displays . . . Y   (Y/N)
  Omission Characters . . . <>
  Autotab Character . . . . +
  Location . . . . . . . . . M   (Beginning, Middle, End)

Select your preference for an entry field delimiter.
3 1. Underscore(USCORE)  2. Reverse video  3. None
Specify color/delimiter to be used when an error is detected.
  Color of field . . . . . .
  Delimiter . . . . . . . . 2 1. Underscore  2. Reverse video  3. None
ISPF Color Attributes.
  Type any color preference for the color attribute you wish to substitute
  Color Attribute You Wish to Substitute
  Blue . . . . . . . . . . .
  Green . . . . . . . . . .
Commands:  RESET HELP END
```

3 Specify the values that should be used when displaying long names:

---

**Note**

The product stores the values for each of the zoom and truncation attributes in your ISPF profile. However, you can also write the values to a user POF.

---

a In the **Allow popup displays** field, specify whether to display a dialog or a panel when the **ZOOM (F4)** key is pressed on an object name:

- If you specify **Y**, the product displays the object name in a dialog. If the name is too long to be displayed in a dialog, the product displays the name in a panel.

- If you specify **N**, the product displays the object name in a panel.

b In the **Omission Characters** field, type the characters that replace the beginning and end of a truncated string in an object name that is too long to be displayed.

c In the **Autotab Character** field, specify whether to display an autotab character in front of an object name that is too long to be displayed:

- If you specify **N**, an autotab character is not displayed and you cannot tab to the object name. To zoom on the object name, position the cursor on the object name and press the **ZOOM (F4)** key.
■ If you specify any character other than N, the character is displayed and you can tab to the object name. To zoom on the object name, press the ZOOM (PF4) key.

Tip
The autotab character indicates which fields must be zoomed to view the entire object name.

d In the Location field, type B, M, or E to specify the location of characters to be omitted in object names that are too long to be displayed.

<table>
<thead>
<tr>
<th>To replace characters</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>At the left end (beginning) of the name</td>
<td>B</td>
</tr>
<tr>
<td>In the middle of the name</td>
<td>M</td>
</tr>
<tr>
<td>At the right end (end) of the name</td>
<td>E</td>
</tr>
</tbody>
</table>

4 Type a value for your entry field delimiter.
5 Type new values for your error message color and delimiter.
6 Type new values for your color preferences.
7 Press END to save your changes and to return to the Main Menu.

Setting the options for data set allocation

Use the Dataset Allocation Options and the Image Copy and Unload Options panels to specify the parameters for allocating your data sets. The worklist and CDL data sets are the only data sets that ALTER and CHANGE MANAGER allocate. All other data sets, such as JCL, Compare, Baseline, Report, and DDL, must be preallocated.

To set the options for data set allocation

1 From the ALTER or CHANGE MANAGER Main Menu, select Options and press Enter.

The ALTER or CHANGE MANAGER Main Options Menu panel is displayed.

2 Select Dataset Allocation Options and press Enter.
The ALTER or CHANGE MANAGER Dataset Allocation Options panel is displayed, as shown in Figure 56 on page 171.

Figure 56: Dataset Allocation Options panel

| ALUFOPT2 ---------- CHANGE MANAGER Dataset Allocation Options ----------------- |
| Command ===> |
| Type information. Then press Enter to continue or PF12 for previous panel. |
| DB2 Subsystem ID .. DEGA |

Specify Dataset Allocation Parameters.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Unitname</th>
<th>Volume</th>
<th>Priqty</th>
<th>Secqty</th>
<th>Alloc Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Worklist/BL Report</td>
<td>SYSDA</td>
<td>15</td>
<td>5</td>
<td></td>
<td>Tracks</td>
</tr>
<tr>
<td>CDL</td>
<td>SYSDA</td>
<td>15</td>
<td>5</td>
<td></td>
<td>Tracks</td>
</tr>
</tbody>
</table>

Commands: HELP END

3 Specify the DB2 subsystem ID (SSID).

4 Specify the data set allocation parameters for a worklist and a baseline report by typing new values for the Unitname, Volume, Priqty, and Secqty fields.

Note

The product stores the values for the data set allocation parameters in your ISPF profile. However, you can also write the values to a user POF.

5 (CHANGE MANAGER only) Specify the data set allocation parameters for the CDL data set by typing new values for the Unitname, Volume, Priqty, and Secqty fields.

Note

The product stores the values for the data set allocation parameters in your ISPF profile. However, you can also write the values to a user POF.

6 Press END to save your changes and to return to the ALTER or CHANGE MANAGER Main Menu.

Setting the processing options for Analysis

Use the Analysis Options panel to specify the data set names, processing options, and job parameters for the Analysis component.

To set the processing options for Analysis

1 From the ALTER or CHANGE MANAGER Main Menu, select Options and press Enter.
2 On the ALTER or CHANGE MANAGER Main Options Menu panel, select **Analysis Options** and press **Enter**.

The ALTER or CHANGE MANAGER Analysis Options panel is displayed, as shown in **Figure 57 on page 172**.

**Figure 57: Analysis Options panel**

<table>
<thead>
<tr>
<th>Command</th>
<th>CHANGED MANAGER Analysis Options</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type information. Then press Enter to continue or PF12 for previous panel.</td>
<td>ALUFOPT3 ---------------- CHANGE MANAGER Analysis Options ----------------</td>
<td>Type ==&gt;</td>
</tr>
<tr>
<td>Specify ANALYSIS dataset names.</td>
<td>JCL . . . . . . 'RDACRJ.V11.ANLYJCL(&amp;WORKID8)'</td>
<td>Run Type . . 2 1. Foreground</td>
</tr>
<tr>
<td>Worklist ... 'RDACRJ.V11.WLBASE01(&amp;WORKID8)'</td>
<td>Diagnostics . . SYSOUT</td>
<td>2. Batch</td>
</tr>
<tr>
<td>Specify default ANALYSIS processing options.</td>
<td>Override Options</td>
<td>Build JCL/Parms</td>
</tr>
<tr>
<td>Run Type . . 2 1. Foreground</td>
<td>Submit/Run</td>
<td>Edit JCL/Parms</td>
</tr>
<tr>
<td>2. Batch</td>
<td>Save Last Used Options</td>
<td>Submit/Run</td>
</tr>
<tr>
<td>Specify default ANALYSIS job parameters.</td>
<td>Include AMS commands</td>
<td>Submit/Run</td>
</tr>
<tr>
<td>Image copy before DROPs or REORGs</td>
<td>Include AMS commands</td>
<td>Submit/Run</td>
</tr>
<tr>
<td>Global AUTHID . .</td>
<td>Include AMS commands</td>
<td>Submit/Run</td>
</tr>
<tr>
<td>Commands: HELP END</td>
<td>Include AMS commands</td>
<td>HELP END</td>
</tr>
</tbody>
</table>

3 Specify the Analysis data set names for the JCL, worklist, and diagnostics.

**Note**
The data sets for the JCL and worklist can be either sequential files or PDS members. The data set for the diagnostics must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground).
The values for these options are stored in your ISPF profile. However, the values can also be written to a user POF.

4 Specify the default processing options for Analysis by typing **S** to select an option.

5 Select whether to run Analysis in foreground or in batch.

6 Specify the default job parameters for Analysis.

**Note**
The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

7 *(optional)* Type a name for a default global AUTHID. For more information about global authorization IDs, see the **ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2**
The product stores the value for this option in your ISPF profile. However, you can also write the value to a user POF.

8 Press END to save your changes and to return to the ALTER or CHANGE MANAGER Main Menu.

Setting the processing options for Import

To define or modify the values in your ISPF profile, you can use the Options panels of ALTER or CHANGE MANAGER.

Use the Import Options panel to specify the data set names and the processing options for the Import component of ALTER or CHANGE MANAGER.

To set the processing options for Import

1 From the ALTER or CHANGE MANAGER Main Menu, select Options and press Enter.

2 On the ALTER or CHANGE MANAGER Main Options Menu panel, select Import Options and press Enter.

The ALTER or CHANGE MANAGER Import Options panel is displayed, as shown in Figure 58 on page 173.

Figure 58: Import Options panel

```
ALUFOPT4 ---------------- CHANGE MANAGER Import Options -----------------------
Command ===> 
Type information. Then press Enter to continue or PF12 for previous panel.
Specify IMPORT dataset names.
  JCL . . . . . 'RDACRJ.V11.IMPJCL(&WORKID8)' 
  Diagnostics . . SYSOUT
Select default IMPORT processing options.
  Run Type . . 2 1. Foreground 
    2. Batch
  Edit CDL 
  $ Build JCL/Parms
  $ Edit JCL/Parms
  - Submit/Run
  - Save Last Used Options

Commands:  HELP END
```

3 Specify the Import data set names for the JCL and diagnostics.
The JCL data set can be either a sequential file or a PDS member. The data set for the Diagnostics file must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground).

The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

4. Specify the default processing options for Import by typing S to select an option.

Note

In ALTER, you can edit the DDL. In CHANGE MANAGER, you can edit the CDL or DDL.

5. Specify whether to run Import in foreground or in batch.

6. Press END to save your changes and to return to the ALTER or CHANGE MANAGER Main Menu.

Setting the processing options for Execution

Use the Execution Options panel to specify the data set names and the method of generating JCL for the Execution component.

To set the processing options for Execution

1. From the ALTER or CHANGE MANAGER Main Menu, select Options and press Enter.

2. On the ALTER or CHANGE MANAGER Main Options Menu panel, select Execution Options and press Enter.
The Execution Options panel is displayed, as shown in Figure 59 on page 175.

Figure 59: Execution Options panel

3 Specify the Execution default data set names for the JCL and diagnostics.

The Execution JCL contains the JCL to run Execution. The batch JCL job contains the JCL to run Batch Execution JCL Generation, which builds the JCL to run Execution.

Note
The data set for JCL can be either a sequential file or a PDS member.

The data set for the Diagnostics file must be a sequential file, unless parallelism is enabled for the Database Administration solution. If you are running a worklist in parallel, the diagnostics file must be allocated to SYSOUT. AEXPRINT is the ddname of the diagnostic output.

The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

4 (optional) Specify up to three Database Request Module (DBRM) libraries that can be concatenated in the Execution JCL for binds in a worklist.

Note
The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

5 Specify the default processing options for Execution by typing S to select an option.

6 Select whether to generate JCL in foreground or in batch.
Setting the processing options for Compare

Use the Compare Options panel to specify the data set names and the processing options for the Compare component of CHANGE MANAGER.

To set the processing options for Compare

1. From the CHANGE MANAGER Main Menu, select **Options** and press **Enter**.

2. On the CHANGE MANAGER Main Options Menu panel, select **Compare Options** and press **Enter**.

The CHANGE MANAGER Compare Options panel is displayed, as shown in Figure 60 on page 176.

**Figure 60: Compare Options panel**

```
ALUF0PT5 ---------------- CHANGE MANAGER Compare Options -----------------------
Command ===>  

Type information. Then press Enter to continue or PF12 for previous panel.

Specify COMPARE dataset names.
CDL . . . . . . 'ACM.DB2V10.V11.CDLBAS01(&WORKID8)'
JCL . . . . . . 'ACM.DB2V10.V11.CMPJCL(&WORKID8)'
Diagnostics . . SYSOUT

- Select to generate Compare report information as comments in CDL

Select default COMPARE processing options.  Run Type . . 2 1. Foreground
$ Override Compare defaults
- Build JCL/Parms
- Edit JCL/Parms
- Submit/Run
- Save Last Used Options

Commands:  HELP END
```

3. Specify the Compare data set names for the CDL, JCL, and diagnostics.

   **Note**

   The data sets for JCL and the CDL file can be either sequential files or PDS members. The data set for the Diagnostics file must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground).

   The product stores the values for the CDL, JCL, and diagnostics data sets in your ISPF profile. However, you can also write the values to a user POF.

4. Specify whether to generate comparison report information as comments in the CDL file.
5 Specify the default processing options for Compare by typing S to select an option.

6 Specify whether to run Compare in foreground or in batch.

7 Press END to save your changes and to return to the CHANGE MANAGER Main Menu.

Setting the processing options for Baseline

Use the Baseline Options panel to specify the data set names and the processing options for the Baseline component of CHANGE MANAGER.

To set the processing options for Baseline

1 From the CHANGE MANAGER Main Menu, select Options and press Enter.

2 On the CHANGE MANAGER Main Options Menu panel, select Baseline Options and press Enter.

The CHANGE MANAGER Baseline Options panel is displayed, as shown in Figure 61 on page 177.

Figure 61: Baseline Options panel

ALUFOPT6 --------------- CHANGE MANAGER Baseline Options ------------------
Command ===> Type information. Then press Enter to continue or PF12 for previous panel.

Specify BASELINE dataset names.
JCL . . . . . . 'RDACRU.V11.BLJCL(&WORKIDB)'
Diagnostics . . SYSOUT

Select default BASELINE processing options. Run Type . . 2 1. Foreground
S Build JCL/Parms 2. Batch
S Edit JCL/Parms
Submit/Run
= Save Last Used Options

Commands: HELP END

3 Specify the Baseline data set names for the JCL and diagnostics.

Note
The JCL data set can be either a sequential file or a PDS member. The data set for the Diagnostics file must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground).
The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

4 Specify the default processing options for Baseline by typing S to select an option.
5 Specify whether to run Baseline in foreground or in batch.

6 Press END to save your changes and to return to the CHANGE MANAGER Main Menu.

**Setting the options for the baseline report**

Use the Baseline Report Options panel to specify the data set names and the processing options for the baseline report for CHANGE MANAGER.

**To set the options for the baseline report**

1 From the CHANGE MANAGER Main Menu, select **Options** and press **Enter**.

2 On the CHANGE MANAGER Main Options Menu panel, select **Baseline Report Options** and press **Enter**.

The CHANGE MANAGER Baseline Report Options panel is displayed, as shown in Figure 62 on page 178.

**Figure 62: Baseline Report Options panel**

ALUFOPT7 ----------- CHANGE MANAGER Baseline Report Options -------------------
Command ===> Type information. Then press Enter to continue or PF12 for previous panel.

Specify BASELINE REPORT dataset names.
JCL . . . . . 'RDACRJ.V11.BLRJCL(&WORKID8)'
Report . . . . 'RDACRJ.V11.BLREPS(&WORKID8)'
Diagnostics . . SYSOUT

Select default REPORT processing options.       Run Type . . 1 1. Foreground
S Build JCL/Parms                                              2. Batch
S Edit JCL/Parms
_ Submit/Run
S Edit Report
_ Save Last Used Options

Commands:  HELP END

3 Specify the baseline report data set names for the JCL, report, and diagnostics.

**Note**
The JCL data set can be either a sequential file or a PDS member. The data set for the Diagnostics file must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground).

The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

4 Specify the default processing options for the baseline report by typing S to select an option.
5 Specify whether to run the baseline report in foreground or in batch.

6 Press END to save your changes and to return to the CHANGE MANAGER Main Menu.

**Setting the user options for CM/PILOT**

The first time that you run CHANGE MANAGER, the Front End copies the values from the installation options module into your ISPF profile.

To define or modify the values in your ISPF profile, you can use the Options panels of the CM/PILOT component of CHANGE MANAGER. Use the panels to specify the task ID data set names, worklist processing options, application profile names, and CHANGE MANAGER data set names.

**To set the user options for CM/PILOT**

1 From the CHANGE MANAGER Main Menu, select **CM/PILOT** and press **Enter**.

2 From the CM/PILOT Main Menu, select **Options** and press **Enter**. The CM/PILOT TASKID Options panel is displayed, as shown in Figure 63 on page 179.

**Figure 63: CM/PILOT TASKID Options panel**

AUCFOPT1 --------------------- CM/PILOT TASKID Options ------------------------

Command ===> Type information. Then press Enter to continue.

Specify TASKID options.

Specify TASKID dataset names.
- Worklist . . . . 'RDACRJ.V11.TASKWL(&TASKID)'
- JCL . . . . . . . 'RDACRJ.V11.TASKJCL(&TASKID)'
- Diagnostics . . . SYSOUT

Select default Worklist processing options.
- Override run options
- Create Worklist
- Edit Worklist
- Create JCL
- Edit JCL
- Submit JCL
- Save Last Used Options

Commands: HELP END CANCEL

3 Specify the task ID data set names for the worklist, JCL, and diagnostics.

These data set names appear as defaults on the TASKID Interface panel.
The JCL data set can be either a sequential file or a PDS member. The data set for the Diagnostics file must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground).

The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

4 Specify the default processing options for the CM/PILOT worklist by typing S to select an option.

5 Press Enter to display the CM/PILOT Application Options panel, which is shown in Figure 64 on page 180.

**Figure 64: CM/PILOT Application Options panel**

```
AUCFOPT2 ------------------ CM/PILOT Application Options ----------------------
Command ==>
Type information. Then press Enter to continue or PF12 for previous panel.
Specify Application profile names.
Type a specific Profile name or type a wildcard pattern for a selection list to be displayed when an Application is being created.
   Example: A.B to choose a specific Profile
            A*.B* for all Profiles like this pattern.
Inbound Migrate Profile
Outbound Migrate Profile
DDL Baseline Profile .
Catalog Baseline Profile
WORKID Name Template .

Commands: HELP END CANCEL
```

6 Specify the profile names and a name template for the work ID. These names appear as defaults on the Create Application panel.

a Type the name of the **Inbound Migrate Profile** that is used by the Import component of CHANGE MANAGER for applying change rules to attributes of data structures that are imported to the local DB2 subsystem.

b Type the name of the **Outbound Migrate Profile** that is used by the Analysis and Compare components of CHANGE MANAGER for generating worklists and CDL files. Outbound migrate profiles can contain change rules, scope rules, and locations.

c Type the name of the **DDL Baseline Profile** that is used by the Baseline component of CHANGE MANAGER for creating baselines of DDL files and worklists.

d Type the name of the **Catalog Baseline Profile** that is used by the Analysis, Baseline, and Compare components of CHANGE MANAGER for specifying the data structures from the DB2 catalog to include in a baseline.
You can specify wildcard patterns for the profiles on the CM/PILOT Application Options panel. Then, when you create an application, the wildcard patterns appear on the Create Application panel. When you press **Enter** on the Create Application panel, a list of profiles is displayed.

e  Type the name of the **WORKID Name Template** that enables you to standardize naming conventions for new work IDs that CM/PILOT creates.

7  Press **Enter** to display the CM/PILOT Output Dataset Options panel, which is shown in Figure 65 on page 181.

**Figure 65: CM/PILOT Output Dataset Options panel**

```
AUCFOPT3 ----------------- CM/PILOT Output Dataset Options -----------------
Command ===> 
Type information. Then press Enter to continue or PF12 for previous panel.

Specify CHANGE MANAGER dataset names.
  JCL ....... 'RDACRJ.V11.EXECJCL(&WORKID8)'
  Worklist ... 'RDACRJ.V11.WLBASE01(&WORKID8)'
  CDL ....... 'RDACRJ.V11.CDLBASE01(&WORKID8)'
```

Commands: HELP END CANCEL

8  Specify the CHANGE MANAGER data set names. These data set names appear as CHANGE MANAGER defaults on the CHANGE MANAGER Datasets panel.

**Note**
You cannot modify the CM/PILOT user options for the CHANGE MANAGER data set names by changing the user options in CHANGE MANAGER. You can modify these user options only from the CM/PILOT Output Dataset Options panel or by refreshing the user options.

The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

a  Type the name of the JCL data set that contains the JCL to process the Analysis worklist.

b  Type the name of the worklist data set that contains the Analysis worklist.

c  Type the name of the CDL data set that contains the CDL which results from a comparison.

9  Press **END** to save your changes and to return to the CM/PILOT Main Menu.

10  Press **END** to return to the CHANGE MANAGER Main Menu.
Setting the JCL options for job cards

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the Jobcard Options panel to specify information about the job cards used in the JCL.

To set the JCL options for job cards

1 Use the following menu selections to display the JCL Generation Jobcard Options Update panel (Figure 66 on page 182):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN Options</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Jobcard Options</td>
</tr>
</tbody>
</table>

Figure 66: JCL Generation Jobcard Options Update panel

```
COMMAND ===> JCL GENERATION JOBCARD OPTIONS UPDATE

Type data and press Enter.

Is a TSO submit exit used to generate jobcards? . N  (Y/N)
Enter Jobcards below:
//MVSCAL2I JOB (&ZACCTNUM),+'&PGMR',
// CLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID
/*
/*
/*
Jcllib . . . . . . ADM.INST110.XXXREXX
Sysexec. . . . . ADM.INST110.XXXREXX
Region size . . . 0M  (See JCL Reference for valid options)
Memlimit. . . . NOLIMIT (See JCL Reference for valid options)
Time parameter . (See JCL Reference for valid options)
System MLIB. . . SYS1.0000.ISPMENU
Runtime HLQ. . . . ADM.INSTXXXX
User HLQ . . . . . . USER.HLQ
LLQ . . . . . . . . . .
ULLQ . . . . . . . . (Leave blank if using runtime enablement)
```

2 In the Is a TSO submit exit used to generate jobcards? field, type Y or N.

3 At Enter Jobcards below, type the job statement information that you want to add to the JCL.

4 In the Jcllib field, type the name of a PDS that contains customized JCL to be included in the job.

5 In the Sysexec field, type the name of the PDS in which a REXX EXEC is a member.
Tip
To indicate the data set name for a different SSID, append the &SSID or &MSSID symbolic variable to the name.

6 In the **Region size** field, type the amount of memory to allocate for each step so that your job can run.

7 In the **Memlimit** field, type the limit on the above-the-bar memory for an address space.

8 In the **Time parameter** field, type the value for the TIME limit for each step in a batch job stream.

9 In the **System MLIB** field, type the name of the system ISPF message library.

10 In the **Runtime HLQ** field, type the high-level qualifier (HLQ) for ISPF data sets for the installation environment.

   During installation, if you chose to use the runtime enablement (RTE) feature, the Installation System set this value to an HLQ for user runtime libraries. If you chose not to use RTE, the Installation System set the value to an HLQ for Execution.

   This value supports the following symbolic variables:

   ■ &DB2V2 and &DB2V3, which resolve to the version of DB2
   ■ &SSID, which resolves to the DB2 subsystem ID

   When you include the &SSID symbolic variable, the product can use a single POF with multiple subsystems.

11 In the **User HLQ** field, type the high-level qualifier used for the user-defined data sets for the installation environment.

   **Note**
   The User HLQ field is used only if the **LLQ** field is blank.

12 In the **LLQ** field, type the low-level qualifier for ISPF data sets for the installation environment.

   During installation, if you chose to use the runtime enablement feature, the Installation System set this value to BMC. If you chose not to use the feature, the Installation System set the value to DB.
13 In the **ULLQ** field, type the low-level qualifier for user-defined data sets for the installation environment.

---

**Note**

If the **User HLQ** field contains a value, the **ULLQ** value is ignored.

---

14 Press END to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.

## Setting the JCL options for STEPLIBs

Use the Options panels of the JCL Generation component to define or modify the values in your ISPF profile and a user POF.

### To set the JCL options

1. Use the following menu selections to display the JCL Generation STEPLIB Options Update panel (Figure 67 on page 184):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Steplib Options</td>
</tr>
</tbody>
</table>

![Figure 67: JCL Generation STEPLIB Options Update panel](image)

2. Specify the data set names for the libraries:
Tip
To indicate the data set name for a different SSID, append the &SSID or &MSSID symbolic variable to the name.

a In the DSNEXIT field, type the data set name of the DB2 EXIT (DSNEXIT) library.

Tip
To indicate the version of DB2, append the &DB2V2 or &DB2V3 symbolic variable to the name.

b In the DB2 DSNLOAD field, type the data set name of the library in which the DB2 DSN Command Processor load modules are stored.

c In the Override lib field, type the data set name for the override LINK library that should appear first in the STEPLIB statement.

d Type the data set names of the LINK libraries for the BMC products.

e In the Additional lib field, type the data set name for the additional LINK library that should appear last in the STEPLIB statement.

3 Press END to save your changes, and to return to the ALTER or CHANGE MANAGER Main Menu.

Setting the JCL options for static data sets

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the Static Data Set Options panel to specify the options for sizing and cleaning up your data sets.

To set the JCL options for static data sets

1 Use the following menu selections to display the JCL Generation Static Data Set Options Update panel:

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
</tbody>
</table>
From this menu | Select this item and press Enter
---|---
JCL Generation Update Main Menu | Static Data Set Options

Figure 68: JCL Generation Static Data Set Options panel

`------------------ JCL GENERATION STATIC DATA SET OPTIONS UPDATE ------------------`
COMMAND ===>
Type data and press Enter.

- Data set sizing option. . . C (N-No Sizing,B-Bmcstats, C-DB2 Catalog,O-Object Sampling)
- Data set sizing device. . . 3390 (3380/3390)
- Max cylinders . . . . . . . 99999 (Do not exceed this primary value in JCL.)
- If max cylinders are exceeded, use the following for DASD data sets
  - Max primary quantity. . . 200 (Cylinders, 1 - 9999)
  - Max secondary quantity. . . 20 (Cylinders, 1 - 9999)
  - Max unit count. . . . . . . . . (Blank or 1 - 59 volumes)
- Include data set cleanup step. . Y (Y/N)
- Return code for cleanup step . . 0 (04)
- Temporary unit . . . . . . . . . SYSDA (SYSDA, SYSALLDA, etc.)
- Include SYSPRIN2 DD. . . . . . . N (Y/N)

Note: See Debugging, Display and Execution Options to display sizing options in the JCL.

2 In the **Data set sizing option** field, type **N**, **B**, **C**, or **O** to specify the sizing method, as shown in Table 19 on page 186.

**Note**

Whether or not data set sizing is performed, DB2 catalog access is required to resolve any symbolic variables. For more information about data set sizing, see “JCL Generation data sets sizing function” on page 159.

Table 19: Data set sizing options

<table>
<thead>
<tr>
<th>For this method</th>
<th>Type</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>No data set sizing</td>
<td>N</td>
<td>The product uses the default primary and secondary quantities for the permanent data sets. To modify the quantities for the primary and secondary space for a permanent data set, see Step 4 on page 188.</td>
</tr>
<tr>
<td>For this method</td>
<td>Type</td>
<td>Considerations</td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
<td>------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Sizing by using statistics from the DASD MANAGER PLUS tables</td>
<td>B</td>
<td>■ The DASD MANAGER PLUS product must be installed and interacting with ALTER or CHANGE MANAGER to use this option.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ If a column is defined as a LONG VARCHAR, this option averages the row size.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ If you select this option for Batch Execution JCL Generation, the DATASETSIZING B keyword is inserted into the AJXIN input stream.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Current BMCSTATS statistics should be available for objects in the worklist before you select this option.</td>
</tr>
<tr>
<td>Sizing by using statistics from the DB2 catalog</td>
<td>C</td>
<td>■ If a column is defined as a LONG VARCHAR, this option does not average the row size.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ If you select this option for Batch Execution JCL Generation, the DATASETSIZING C keyword is inserted into the AJXIN input stream.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Current IBM RUNSTATS catalog statistics should be available for objects in the worklist before you select this option.</td>
</tr>
<tr>
<td>Estimating sizes based on physical, random sampling of VSAM data sets</td>
<td>O</td>
<td>■ If a column is defined as a LONG VARCHAR, this option averages the row size.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ If you select this option for Batch Execution JCL Generation, the DATASETSIZING O keyword is inserted into the AJXIN input stream.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ You can use this option if the statistical information in the DB2 catalog or in the DASD MANAGER PLUS tables is not current. However, JCL generation might take additional time to complete.</td>
</tr>
</tbody>
</table>

You can specify whether to include comments in the Execution JCL that show statistics for determining data set sizes. For more information, see “Setting the JCL debugging display and execution options” on page 205.

3 In the **Data set sizing device** field, specify the type of DASD to use in calculating the sizes of data sets.
4 In the **Max cylinders** field, specify the maximum number of cylinders for a data set.

5 Specify the values that should be used for the following options when the maximum number of cylinders are exceeded:

a In the **Max primary quantity** field, type the number of cylinders for the maximum primary quantity.

b In the **Max secondary quantity** field, type the number of cylinders for the maximum secondary quantity.

c In the **Max unit count** field, type the maximum number of volumes.

If you want to use multiple data sets on DASD, specify a value greater than 1 for **Max unit count**. On the JCL Generation Individual Data Set Options Update panels (see “Setting the JCL options for permanent data sets” on page 195), specify the name of a DASD unit.

6 In the **Include data set cleanup step** field, type **Y** or **N** to specify whether to generate a step in the JCL to delete the permanent work data sets. The product deletes the data sets at the end of Execution. For Batch Execution JCL Generation, selecting this option inserts the JCLCLEANUP YES keyword into the AJXIN input stream.

You can generate the JCL for a job step that automatically deletes many of the permanent work data sets that Execution creates. Execution creates these data sets during worklist processing and sets the disposition of the data sets to new, catalog, catalog (DISP=(NEW,CATLG,CATLG)).

The cleanup job step is performed only if the condition code returned from any previous job step is less than or equal to the number that is specified in the **Return code for cleanup step** field. Table 20 on page 188 lists the types of work data sets that are included in the cleanup job step. These data sets are automatically deleted unless otherwise noted.

---

### Table 20: Work data sets in the JCL cleanup job step

<table>
<thead>
<tr>
<th>Work data set</th>
<th>ddbname</th>
<th>Used in JCL cleanup by</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discard (SYSDISC) 6.a</td>
<td>SYSDnnnn</td>
<td>LOADPLUS IBM LOAD</td>
</tr>
<tr>
<td>Error</td>
<td>SYSERnnn</td>
<td>CHECK PLUS LOADPLUS IBM CHECK DATA IBM LOAD</td>
</tr>
</tbody>
</table>
The unload data sets that are used by REORG PLUS and IBM REORG are deleted automatically.

**Table 21 on page 189** lists the work data sets that are used by the corresponding utilities.

**Table 21: Work data sets used by utilities**

<table>
<thead>
<tr>
<th>Work data set</th>
<th>Utility</th>
</tr>
</thead>
<tbody>
<tr>
<td>SORTOUT</td>
<td>CHECK PLUS</td>
</tr>
<tr>
<td>SORPnnn</td>
<td>LOADPLUS</td>
</tr>
<tr>
<td>SORTOnnn</td>
<td>REORG PLUS</td>
</tr>
<tr>
<td>SYSUTnnn</td>
<td>IBM CHECK DATA</td>
</tr>
<tr>
<td>SUTnnn</td>
<td>IBM LOAD</td>
</tr>
<tr>
<td>WRKnnn</td>
<td>IBM REORG</td>
</tr>
<tr>
<td></td>
<td>IBM RECOVER INDEX</td>
</tr>
<tr>
<td></td>
<td>IBM REBUILD INDEX</td>
</tr>
</tbody>
</table>
7 In the **Return code for cleanup step** field, specify the value that should be compared against the condition code that is returned from any previous job step.

You can specify any two-digit value for the return code; however, BMC recommends that you specify 4.

8 In the **Temporary unit** field, type the name of the unit that is used to allocate temporary files when JCL is generated.

The value of the unit can be a name from 1 to 8 characters long, blank, or NONE. To omit the UNIT parameter from the JCL, specify NONE.

9 In the **Include SYSPRIN2 DD** field, type Y or N to specify whether to generate the `//SYSPRIN2 DD SYSOUT=*` DD in the JCL.

The SYSPRIN2 output data set contains SYSPRINT output messages for versions 10.2 and later of the following BMC utilities:

- CHECK PLUS
- LOADPLUS
- REORG PLUS
- UNLOAD PLUS

If you type Y, you can view the SYSPRINT output from a utility while an execution job runs the utility or when an execution job cancels during the running of the utility.

---

**Note**

SYSPRIN2 data sets have the following restrictions:

- When you specify BMCSTATS YES or UPDATEDB2STATS YES for LOADPLUS or REORG PLUS, SYSPRIN2 does not contain the statistics report from the Common Statistics component.

- When invoking the IBM DSNUTILB utility, REORG PLUS and LOADPLUS ignore the SYSPRIN2 DD statement.

10 Press END to save your changes, and to return to the ALTER or CHANGE MANAGER Main Menu.
Setting the JCL options for tapes

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the Tape Options panel to specify information about tape units and stacking options.

To set the JCL options for tapes

1. Use the following menu selections to display the JCL Generation Tape Options Update panel (Figure 69 on page 191):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Tape Options</td>
</tr>
</tbody>
</table>

![Figure 69: JCL Generation Tape Options Update panel](image)

2. In the Tape Unit 1, Tape Unit 2, and Tape Unit 3 fields, type the names of valid tape units for your installation.

3. In the Tape Volume count field, type the maximum number of tape volumes.

4. In the Tape EXPDT field, type the expiration date for a tape.

5. In the Tape RETPD field, type the retention date for a tape.

6. In the Tape TRTCH field, type the parity, data conversion, translation, and compression value for 7-track tape drives as shown in Table 22 on page 192.
Table 22: Values for 7-track tape drives

<table>
<thead>
<tr>
<th>To choose</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not to use seven-track tape drives</td>
<td>(Leave the field blank)</td>
</tr>
<tr>
<td>Odd parity, conversion, and no translation</td>
<td>C</td>
</tr>
<tr>
<td>Even parity, no conversion, and no translation</td>
<td>E</td>
</tr>
<tr>
<td>Odd parity, no conversion, and translation</td>
<td>T</td>
</tr>
<tr>
<td>Even parity, no conversion, and translation</td>
<td>ET</td>
</tr>
<tr>
<td>Data compression</td>
<td>COMP</td>
</tr>
<tr>
<td>No data compression</td>
<td>NOCOMP</td>
</tr>
</tbody>
</table>

For each type of copy or product data set, type Y or N to specify whether the data sets should be stacked on a tape with data sets of the same type.

Consider the following items if you choose tape stacking:

- Tape stacking options for baseline recovery data sets apply to CHANGE MANAGER only.
- Tape stacking options for backup SYSREC and archive data sets apply to CATALOG MANAGER only.
- Tape stacking is not applicable if you choose to dynamically allocate copy or unload data sets, unless you are using the COPY PLUS utility to copy explicitly created table spaces.
- Tape stacking is not applicable if you use the Database Administration solution to execute a worklist in parallel.
- Tape stacking is disabled in the JCL for a worklist if all of the following conditions exist:
  - You use tape for unload (SYSREC) data sets, or you use tape because the maximum threshold value for DASD for a data set is exceeded.
  - You use tape stacking for unload (SYSREC) data sets.
  - You use the UNLOAD PLUS and LOADPLUS utilities and you include partitioned table spaces in the scope.

If you are using the UNLOAD PLUS and LOADPLUS utilities to multitask the unloading and loading of data, you cannot use tape stacking for unload (SYSREC) data sets. If you run the JCL, you might exceed the number of tape drives at your site.
8 Press END to save your changes, and to return to the ALTER or CHANGE MANAGER Main Menu.

**Setting the JCL options for temporary work data sets**

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the Options For Sort Files panel to specify information about the temporary work data sets:

- **SORTWORK (SORTWK)**
- **DATAWORK (DATAWK)**
- **LOGSORT (LOGSWK)**

The temporary work data sets (such as SORTWORK) are defined by using `DISP=(*PASS)` in the JCL.

**To set the JCL options for temporary work data sets**

1 Use the following menu selections to display the JCL Generation Options For Sort Files Update panel (*Figure 70 on page 194*):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>Menu</td>
<td></td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Individual Data Set Options</td>
</tr>
</tbody>
</table>
2 Specify the options for SORTWORK data sets:

   a In the Number of SORTWORK Data Sets field, type the number of SORTWORK data sets.

   b In the SORTWORK unit name field, type the name of the unit for SORTWORK data sets.

      The value of the unit can be a name from 1 to 8 characters long, blank, or NONE. To omit the UNIT parameter from the JCL, specify NONE.

3 Specify the options for DATAWORK data sets:

   a In the Number of DATAWORK Data Sets field, type the number of DATAWORK data sets.

   b In the DATAWORK unit name field, type the name of the unit for DATAWORK data sets.

      The value of the unit can be a name from 1 to 8 characters long, blank, or NONE. To omit the UNIT parameter from the JCL, specify NONE.

4 Specify the options for LOGSORT data sets:

   a In the Number of LOGSORT Data Sets field, type the number of LOGSORT data sets.
b In the **LOGSORT unit name** field, type the name of the unit for LOGSORT data sets.

The value of the unit can be a name from 1 to 8 characters long, blank, or NONE. To omit the UNIT parameter from the JCL, specify NONE.

5 If you typed **N** in the **Data set sizing option** field in Step 2 on page 186, specify the default primary and secondary quantities:

a In the **Default Primary Quantity** field, type the value for the primary quantity in cylinders.

b In the **Default Secondary Quantity** field, type the value for the secondary quantity in cylinders.

6 Specify the IBM Storage Management Subsystem (SMS) definitions for the optional SORTOUT data set classes:

a In the **SMS Data Class** field, type the name of the data class.

b In the **SMS Storage Class** field, type the name of the storage class.

c In the **SMS Management Class** field, type the name of the management class.

7 In the **$SORTPARM data set name** field, type the name of the data set that provides parameters for SyncSort.

8 Press END to save your changes, and to return to the ALTER or CHANGE MANAGER Main Menu.

### Setting the JCL options for permanent data sets

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the individual data set options panels to specify information about the following permanent work data sets and image copy data sets:

- **Sortout (WORKDDN or LOADDN)**
- **Sysut (WORKDDN)**
- **Copy (COPYDDN, RECOVERYDDN, RECOVERDDN, ICDDN, RECOVERYICDDN, OUTCOPYDDN, FCOPYDD, or EXPORTDDN)**
- **Sysrec (UNLDDN, INDDN, or UNLOADDDN)**
- Archive (ARCHDDN)
- Cntl file (CNTLDDN)
- *(CHANGE MANAGER only)* Baseline recovery
- Discard (DISCARDDN)
- Error (ERRDDN)
- Map (MAPDDN)
- Report
- Punch (PUNCHDDN)
- Filter (FILTERDDN)

The permanent work data sets that contain data allow restarts. They are defined by using DISP=(NEW,CATLG) or DISP=SHR for restart or startover JCL. Examples include input (SYSUT), output (SORTOUT), discard (SYSDISC), map (SYSMAP), error (SYSERR), and punch (SYSPUNCH).

Other permanent data sets are used for restart and recover purposes. They use the same dispositions as the permanent work data sets. Examples include unload (SYSREC), copy (SYSCOOPY), and baseline recovery (BLRP) *(for CHANGE MANAGER only)*.

The ROWID and LOB SYSREC data sets are used only by the UNLOAD PLUS utility and the LOB DATA MOVER program in the Database Administration solution to unload and load data contained in a ROWID column and LOB columns.

**To set the JCL options for permanent data sets**

1. Use the following menu selections to display the panels for permanent work data sets:

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Individual Data Set Options</td>
</tr>
<tr>
<td>JCL Generation Individual Data Set Options</td>
<td>Type of data set (SORTOUT, SYSUT, or COPY)</td>
</tr>
</tbody>
</table>
Not all of the options are available on all of the data set options panels. If you choose to dynamically allocate your copy or unload data sets, any changes that you make to the copy (SYSCOPY), unload (SYSREC), and baseline recovery (BLRP) (for CHANGE MANAGER only) data set options in the JCL Generation override panels do not take effect.

Figure 71: JCL Generation Data Set Options For Sortout Update panel

<table>
<thead>
<tr>
<th>COMMAND ====&gt;</th>
<th>JCL GENERATION DATA SET OPTIONS FOR SORTOUT UPDATE ---</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter Data Set Prefix below:</td>
<td>. &amp;PREFIX..&amp;WKID..&amp;STEPN</td>
</tr>
<tr>
<td>Unit Name . . . . . . . . . . . SYSDA (SYSDA, TAPE, etc)</td>
<td></td>
</tr>
<tr>
<td>Primary Space . . . . . . . . . 10 (Cylinders)</td>
<td></td>
</tr>
<tr>
<td>Secondary Space . . . . . . . . . 2 (Cylinders)</td>
<td></td>
</tr>
<tr>
<td>Tape EXPDT. . . . . . . . . . . . (Blank or YYDDD or YYYY/DDD)</td>
<td></td>
</tr>
<tr>
<td>Tape RETPD. . . . . . . . . . . . (Blank or 1 - 9999 days)</td>
<td></td>
</tr>
<tr>
<td>SMS Data Class. . . . . . . . . (Blank or Data Class)</td>
<td></td>
</tr>
<tr>
<td>SMS Storage Class . . . . . . . (Blank or Storage Class)</td>
<td></td>
</tr>
<tr>
<td>SMS Management Class . . . . (Blank or Management Class)</td>
<td></td>
</tr>
<tr>
<td>Threshold Value . . . . . . . . 0 (Cylinders, 0 means no Threshold)</td>
<td></td>
</tr>
<tr>
<td>Alternate Unit Name . . . . . . (SYSDA, TAPE, etc)</td>
<td></td>
</tr>
<tr>
<td>Alternate SMS Data Class . . . (Blank or Data Class Name)</td>
<td></td>
</tr>
<tr>
<td>Alternate SMS Storage Class . . (Blank or Storage Class Name)</td>
<td></td>
</tr>
<tr>
<td>Alternate SMS Management Class (Blank or Management Class Name)</td>
<td></td>
</tr>
</tbody>
</table>

2 Specify the prefix for the data set.

Consider the following items when you specify the prefix:

- The suffix in the name of an unload (SYSREC) or image copy data set varies, depending on whether the data set is dynamically allocated:

  - For data sets that are not dynamically allocated, JCL Generation appends the ddname to the prefix to create the name of the data set.
  
  To suppress the ddname, specify Y for the appropriate keyword in the POF.

<table>
<thead>
<tr>
<th>Data set</th>
<th>POF keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local primary copy</td>
<td>PCPY1_SUPPRESS_SUFF</td>
</tr>
<tr>
<td>Local backup copy</td>
<td>PCPY2_SUPPRESS_SUFF</td>
</tr>
<tr>
<td>Recovery primary copy</td>
<td>RCPY1_SUPPRESS_SUFF</td>
</tr>
<tr>
<td>Recovery backup copy</td>
<td>RCPY2_SUPPRESS_SUFF</td>
</tr>
<tr>
<td>File reference</td>
<td>UNLD_FREF_SUPPR_SUFF</td>
</tr>
<tr>
<td>Primary SYSREC</td>
<td>UNLD1_SUPPRESS_SUFF</td>
</tr>
</tbody>
</table>
Data set | POF keyword
---|---
Backup SYSREC | UNLD2_SUPPRESS_SUFF
(UNLOAD PLUS in the Database Administration solution) ROWID SYSREC | UNLD3_SUPPRESS_SUFF
(LOB DATA MOVER program in the Database Administration solution) LOB SYSREC | UNLD4_SUPPRESS_SUFF

—For data sets that are dynamically allocated, Analysis appends a period, a two-character abbreviation, and a six-digit number to the prefix to create the name of the data set.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline recovery</td>
<td>&amp;PREFIX..&amp;OBNOD.BLnnnnnn</td>
</tr>
<tr>
<td>Discard</td>
<td>&amp;PREFIX..&amp;OBNOD.SDnnnnnn</td>
</tr>
<tr>
<td>Local primary copy</td>
<td>&amp;PREFIX..&amp;OBNOD.LPnnnnnn</td>
</tr>
<tr>
<td>Local backup copy</td>
<td>&amp;PREFIX..&amp;OBNOD.LBnnnnnn</td>
</tr>
<tr>
<td>Recovery primary copy</td>
<td>&amp;PREFIX..&amp;OBNOD..P&amp;PART.RPnnnnnn</td>
</tr>
<tr>
<td>Recovery backup copy</td>
<td>&amp;PREFIX..&amp;OBNOD..P&amp;PART.RBnnnnnn</td>
</tr>
<tr>
<td>EXPORT data set</td>
<td>&amp;PREFIX..&amp;WKID.EXnnnnnn</td>
</tr>
<tr>
<td>Punch</td>
<td>&amp;PREFIX..&amp;WKID..&amp;STEPN.PUnnnnnn</td>
</tr>
<tr>
<td>SYSUT</td>
<td>&amp;PREFIX..&amp;WKID..&amp;STEPN.SUnnnnnn</td>
</tr>
<tr>
<td>Primary SYSREC</td>
<td>&amp;USERID..&amp;MSSID..&amp;WORKID8.SRnnnnnn</td>
</tr>
<tr>
<td>ROWID SYSREC</td>
<td>&amp;USERID..&amp;MSSID..&amp;WORKID8.SRnnnnnn</td>
</tr>
<tr>
<td>LOB SYSREC</td>
<td>&amp;USERID..&amp;MSSID..&amp;WORKID8.SRnnnnnn</td>
</tr>
<tr>
<td>XML file reference</td>
<td>&amp;USERID..&amp;MSSID..&amp;WORKID8.SRnnnnnn.Xnn</td>
</tr>
<tr>
<td>LOB file reference</td>
<td>&amp;USERID..&amp;MSSID..&amp;WORKID8.SRnnnnnn.Lnn</td>
</tr>
</tbody>
</table>
The EXPORT data set is used by the Copy Migration feature of the Recovery Management and Database Administration solutions. The Copy Migration feature includes the COPY PLUS EXPORT command and RECOVER PLUS IMPORT command.

If you are using the UNLOAD PLUS and LOADPLUS utilities to separate data sets for each table space partition and multitask the unloading and loading of data, Analysis also appends the P&PART variable to the prefix to create the name of the primary SYSREC data set (&USERID..&MSSID..&WORKID8.SRnnnnnnn.P&PART).

For the ROWID SYSREC data set (ROWIDDSN), Analysis also appends .RWID to the prefix (&USERID..&MSSID..&WORKID8.SRnnnnnnn.RWID). The UNLOAD PLUS utility dynamically allocates the ROWID SYSREC data set.

For the LOB SYSREC data set (LOBDSN), Analysis also appends .Cnnn to the prefix, where nnn is the number of the LOB column in the table (&USERID..&MSSID..&WORKID8.SRnnnnnnn.Cnnn). The LOB DATA MOVER program dynamically allocates the LOB SYSREC data set.

The file reference data set is used in the SYXCnn OUTPUT descriptor for the BMC UNLOAD PLUS utility and in the SYXCnn TEMPLATE descriptor for the IBM UNLOAD utility. The utilities dynamically allocate the XML file reference data set. Assume that you are using the unload and load utilities to separate data sets for each table space partition and to multitask unloading and loading data. In that case, Analysis also appends the P&PART variable to the prefix to create the name of the primary SYSREC data set (&USERID..&MSSID..&WORKID8.SRnnnnnnn.Xnn.P&PART).

The file reference data set is used in the SYXBnn OUTPUT descriptor for the BMC UNLOAD PLUS utility and in the SYLBnn TEMPLATE descriptor for the IBM UNLOAD utility. The utilities dynamically allocate the LOB file reference data set. Assume that you are using the unload and load utilities to separate data sets for each table space partition and to multitask unloading and loading data. In that case, Analysis also appends the P&PART variable to the prefix to create the name of the primary SYSREC data set (&USERID..&MSSID..&WORKID8.SRnnnnnnn.Lnn.P&PART).

**Note**

You must ensure the uniqueness of the name of the data set.
To specify a GDG for the local and recovery image copy data sets, add the &GDG symbolic variable to the end of the data set prefix (Figure 72 on page 200).

Figure 72: Using the &GDG symbolic variable

```
----------- JCL GENERATION DATA SET OPTIONS FOR LOCAL PRIMARY COPY UPDATE ----
COMMAND ==>

Type data and press Enter. Press PF3 or END to return to the main panel.

Enter Data Set Prefix below:
&PREFIX..&OBNOD(&GDG)

Unit Name .................. SYSDA (SYSDA, TAPE, etc)
Primary Space ............. 10 (Cylinders)
Secondary Space .......... 2 (Cylinders)
Tape EXPDT. ............... (Blank or YYDDD or YYYY/DDD)
Tape RETPD. ............... (Blank or 1 - 9999 days)
SMS Data Class. .......... (Blank or Data Class)
SMS Storage Class ...... (Blank or Storage Class)
SMS Management Class ... (Blank or Management Class)

Threshold Value .......... 0 (Cylinders, 0 means no Threshold)
Alternate Unit Name ...... (SYSDA, TAPE, etc)
Alternate SMS Data Class (Blank or Data Class Name)
Alternate SMS Storage Class (Blank or Storage Class Name)
Alternate SMS Management Class (Blank or Management Class Name)
```

When you use the &GDG variable, JCL Generation resolves the data set name using the symbolic variable, and the name includes the GDG number (Figure 73 on page 200).

Figure 73: Data set names resolved with the &GDG symbolic variable

```
/* ---------------- U N I T Y  C O P Y DD S T A T E M E N T S */
SYCLO001 DD DSN=RDACRJ.DEHOMCJ.S9(+1),
          DCB=(SYS1.MODEL),
          DISP=(NEW,CATLG,CATLG),
          SPACE=(CYL,(10,2),RLSE),
          UNIT=SYSDA
SYCLO002 DD DSN=RDACRJ.DEHOMCJ.S3(+1),
          DCB=(SYS1.MODEL),
          DISP=(NEW,CATLG,CATLG),
          SPACE=(CYL,(10,2),RLSE),
          UNIT=SYSDA
SYCLO003 DD DSN=RDACRJ.DEHOMCJ.S2(+1),
          DCB=(SYS1.MODEL),
          DISP=(NEW,CATLG,CATLG),
          SPACE=(CYL,(10,2),RLSE),
          UNIT=SYSDA
SYCLO004 DD DSN=RDACRJ.DEHOMCJ.S11(+1),
          DCB=(SYS1.MODEL),
          DISP=(NEW,CATLG,CATLG),
          SPACE=(CYL,(10,2),RLSE),
          UNIT=SYSDA

/* ---------------- S O R T  W O R K DD S T A T E M E N T S */
SORTWK01 DD UNIT=SYSDA,
          SPACE=(CYL,(10,2)),
          DISP=(NEW,DELETE)
SORTWK02 DD UNIT=SYSDA,
```
If you specify a GDG for a dynamically allocated data set, Analysis does not append the two-character abbreviation and the six-digit number to the prefix.

---

**Note**

ALTER and CHANGE MANAGER use data set prefixes for unload (SYSREC) and image copy data sets. These prefixes ensure the uniqueness of the name of the data set when the data set is dynamically allocated. The POF that is generated in the HLQ.UBMCCNTL data set includes the new prefixes, unless you copy an existing POF during installation. If you copy an existing POF during installation to the HLQ.UBMCCNTL data set, you might need to modify the data set prefixes.

- The `SRnnnnnnn` unload data set can contain the `&MSSID` symbolic variable, which identifies the DB2 subsystem ID (SSID) on the sending subsystem for a two-phase migration. The Analysis component also generates the `-JCLP` worklist command with the `&MSSID` variable and a value that refers to the sending SSID.

- The prefixes for copy data sets can contain the `&TSSID` symbolic variable, which identifies the receiving (or target) DB2 subsystem ID (SSID). If you specify the name of the DB2 SSID as the target SSID for a migrate-type work ID, the Analysis component generates a `-JCLP` worklist command with the TSSID parameter. The TSSID parameter specifies the name of the receiving (or target) SSID. JCL Generation uses the name of the receiving SSID to build the Execution JCL. For example, if the following command is in the worklist, JCL Generation uses the DEBA SSID for the receiving subsystem:

  ```
  -JCLP 000700 MIGR TSSID DEBA
  ```

If you do not specify the name of an SSID as the target, or you specify an asterisk, JCL Generation uses the value of the `&SSID` symbolic variable as the value for the `&TSSID` variable.

---

**Note**

If the copy data set is dynamically allocated, JCL Generation substitutes the `&SSID` variable for the `&TSSID` variable in the COPY OUTPUT descriptor. The copy utility resolves the `&SSID` variable.
To specify the target SSID, see the task for defining migrate options for a migrate-type work ID in the *ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.*

### Setting options

3. In the **Unit Name** field, type the name of the unit.

   Consider the following items when you specify the unit:

4. If you typed **N** in the **Data set sizing option** field in Step 2 on page 186, specify the default primary and secondary quantities:

   a. In the **Primary Space** field, type the value for the primary quantity in cylinders.

   b. In the **Secondary Space** field, type the value for the secondary quantity in cylinders.

   This option is not applicable if you choose to dynamically allocate copy or unload data sets.

5. In the **Tape EXPDT** field, type the expiration date for a tape.

6. In the **Tape RETPD** field, type the retention date for a tape.

7. Specify the SMS definitions for the data set classes:

   a. In the **SMS Data Class** field, type the name of the data class.

   b. In the **SMS Storage Class** field, type the name of the storage class.

   c. In the **SMS Management Class** field, type the name of the management class.

8. In the **Threshold Value** field, type, in cylinders, the primary quantity for the data set.

   If this value is exceeded, JCL Generation uses the alternate unit and the alternate SMS parameters. Zero indicates that a threshold is not specified for the unit. If you specify zero, JCL Generation does not use an alternate unit and the alternate SMS parameters.

When ALTER or CHANGE MANAGER dynamically allocates the copy data sets for IBM COPY, it uses TEMPLATE descriptors in the worklist for local and recovery primary and backup copies. If you specify a value for Threshold Value, the product provides templates for a primary set of data set attributes and an alternate set of data set attributes. If the threshold is exceeded, the utility can perform template switching and use the alternate data set.
For more information about the TEMPLATE descriptors, see ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2 and the -COPY worklist command in the ALTER and CHANGE MANAGER for DB2 Reference Manual.

---

**Note**

If you are processing a worklist in parallel, do not specify a threshold value.

---

9 In the *Alternate Unit Name* field, type the alternate name of the unit to be used if the threshold value specified in Step 8 on page 202 is exceeded.

---

**Note**

The alternate unit name must be a valid tape unit name. For more information, see “Setting the JCL options for tapes” on page 191.

---

10 Specify the SMS definitions for the alternate data set classes:

- In the *Alternate SMS Data Class* field, type the name of the data class.
- In the *Alternate SMS Storage Class* field, type the name of the storage class.
- In the *Alternate SMS Management Class* field, type the name of the management class.

11 Press END to save your changes, and to return to the ALTER or CHANGE MANAGER Main Menu.

---

**Setting the JCL generation data group options**

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the Generation Data Group Options panel to specify information about GDGs.

**To set the JCL options for generation data groups**

1 Use the following menu selections to display the JCL Generation Generation Data Group Options Update panel (Figure 74 on page 204):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>From this menu</td>
<td>Select this item and press Enter</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Generation Data Group Options (GDGDs)</td>
</tr>
</tbody>
</table>

### Figure 74: JCL Generation Generation Data Group Options Update panel

```
COMMAND ===>
Type data and press Enter.
Define GDG base at JCL generation?.. N (Y/N)
Specify NSCR on GDG definition?.. N (Y/N)
Number of primary copy GDG entries.. 10 (1-255)
Number of recovery copy GDG entries.. 10 (1-255)
Type GDG Model data set below:
... SYS1.MODEL
NOTE: GDGs only apply to copy data sets.
```

2. In the **Define GDG base at JCL generation?** field, type **Y** or **N** to specify whether JCL Generation creates the base of the GDG.

   **Note**

   JCL Generation cannot create the base of the GDG if you use IBM COPY to generate image copies for implicitly created objects.

3. In the **Specify NSCR on GDG definition?** field, type **Y** if the base of a GDG is defined in the IDCAMS DEFINE command as EMPTY (NSCR), or **N** if the base is defined as SCRATCH (SCR):

   - SCR (the default) indicates to scratch (delete) the generation data set when it is uncataloged.
   - NSCR indicates to uncatalog the generation data set when the maximum number of generation data sets to keep is reached.

4. In the **Number of primary copy GDG entries** field, type the maximum number of generation data sets to keep for primary copies.

5. In the **Number of recovery copy GDG entries** field, type the maximum number of generation data sets to keep for recovery copies.

6. At **Type GDG Model data set below**, type the name of the GDG model data set.

   **Note**

   If you type NONE, the DCB=model.dataSetName is omitted from the JCL for the data set.

7. Press END to save your changes, and to return to the ALTER or CHANGE MANAGER Main Menu.
Setting the JCL debugging display and execution options

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the Debugging, Display and Execution Options panel to specify information about how comments are handled in the JCL.

To set the JCL options for debugging, display, and Execution

1. Use the following menu selections to display the JCL Generation Debugging, Display And Execution Options Update panel:

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Debugging, Display and Execution Options</td>
</tr>
</tbody>
</table>

![Figure 75: JCL Generation Debugging, Display And Execution Options Update panel](image)

AJX0DBG JCL GENERATION DEBUGGING,DISPLAY AND EXECUTION OPTIONS UPDATE ---------
COMMAND ===> 

Type data and press Enter.

Include data set sizing comments in JCL . . . . . . . . . . . . N (Y/N)
Include variable substitution comments in JCL . . . . . . . . N (Y/N)
Suppress comments in JCL . . . . . . . . . . . . . . . . . . . . . . N (Y/N)
NOTE: Do not set suppress comments to Y if you have specified either sizing or variable substitutions.
Specify an Alternate Program for IKJEFT01 . . . . . . . . . .
Specify a Plan name to run DSNTIAD . . . . . . . . . . . . . .
Pre Job Step JCL INCLUDE member name . . . . . . . . . . . .
Post Step JCL INCLUDE member name . . . . . . . . . . . . . .
Post Job JCL INCLUDE member name . . . . . . . . . . . . . .
Include in AEXIN parameters:
SYNCCDELETE . . . . N (Y/N) BINDFAIL . . . . . . N (Y/N)
HASHFAIL . . . . N (Y/N) HASHWARNRC . . . (NUMERIC)
REBINDFAIL . . . N (Y/N) REBINDRC . . . . (NUMERIC)
2MEGSOL . . . . . N (Y/N) NOFAILNOIMAGECOPY N (Y/N)
STOPWAIT . . . . 3 (NUMERIC)
STOPWAIT SECS . . 10 (NUMERIC)

2. Specify whether to include debugging comments in the generated JCL:

**Note**

BMC recommends that you include the comments if you suspect that the JCL was generated incorrectly and you need to send documentation to Customer Support. If you want to reduce the number of lines of JCL, do not include the comments.

a. In the Include data set sizing comments in JCL field, type Y or N to specify whether to include comments in the generated JCL that show statistics for determining data set sizes.
Comments are shown as dsso/ cc, where dsso is the data set sizing option and cc is a comment code. Table 23 on page 206 lists the comment codes that Execution generates in the JCL.

Table 23: Comment codes for data set sizing

<table>
<thead>
<tr>
<th>Data set sizing options</th>
<th>Comment code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>B, C, or O</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td>Uses statistics from the DB2 catalog</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>Uses the high relative-byte address (RBA)</td>
</tr>
<tr>
<td>M</td>
<td></td>
<td>Uses multiple objects to size one data set (for example, SYSUTs)</td>
</tr>
<tr>
<td>N</td>
<td></td>
<td>Indicates that the data set could not be sized because statistics could not be found</td>
</tr>
<tr>
<td>O</td>
<td></td>
<td>Uses VSAM object sampling</td>
</tr>
<tr>
<td>S</td>
<td></td>
<td>Uses the following formula to calculate the SORTWK size: ((work\ space \times 2) / \text{number of SORTWK data sets})</td>
</tr>
<tr>
<td>W</td>
<td></td>
<td>Warns that the sizing might be inaccurate</td>
</tr>
</tbody>
</table>

Figure 76 on page 206 shows example comments in the Execution JCL.

**Figure 76: Execution JCL with comments**

```/* Errddn output dd statements */
/* N = Can't size data set for dd SYSER001, defaults used because no BMCstats were found for tb ACMX01.T_X01PS */
//SYSER001 DD DSN=RDAMCG3.MG1217D.STEP1.SYSER001,
//     DISP=(NEW,CATLG,CATLG),
//     SPACE=(CYL,(10,2),RLSE), ESTIMATE=B/N
// UNIT=SYSDA

/* Sort work dd statements */
/* S = Sortwk size is \((work\ space \times 2) / \#\text{SORTWKS}\) */
//SORTWK01 DD UNIT=SYSDA,
//     SPACE=(CYL,(1,1)), ESTIMATE=C/S
// DISP=(NEW,DELETE)
```

b In the **Include variable substitution comments in JCL** field, type Y or N to specify whether to include comments that show the SLIB variables and their assigned values.

JCL Generation uses these variables to resolve the names of the data sets in the generated JCL.

3 In the **Suppress comments in JCL** field, type Y or N to specify whether to suppress all comments in the generated JCL.
Note
If you chose to include either the statistics comments or the variable comments in Step 2 on page 205, you cannot select to suppress all comments in the JCL.

4 In the Specify an Alternate Program for IKJEFT01 field, type the name of a program to be used instead of IKJEFT01.

Note
The alternate program is only used for nonworklist JCL.

5 In the Specify a Plan name to run DSNTIAD field, type the name of the DB2 plan to run the IBM DSNTIAD program.

6 In the Pre Job Step JCL INCLUDE member name field, type the name of a JCL member to be included before each step in the JCL.

7 In the Post Step JCL INCLUDE member name field, type the name of a JCL member to be included after each step in the JCL.

8 In the Post Job JCL INCLUDE member name field, type the name of a JCL member to be included at the end of a job.

9 For each AEXIN keyword, type Y or N to specify whether to include the keyword in the AEXIN input stream.

For more information about the keywords, see the list of keywords in the ALTER and CHANGE MANAGER for DB2 Reference Manual.

10 Press END to save your changes, and to return to the ALTER or CHANGE MANAGER Main Menu.

Setting the JCL utility installation options module name options

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the BMC Utility Option Module Names panel to specify the name of the installation options module for the BMC utilities.

To set the JCL options for utility installation options module names

1 Use the following menu selections to display the JCL Generation BMC Utility Option Module Names Update panel (Figure 77 on page 208):
From this menu | Select this item and press Enter
--- | ---
ALTER or CHANGE MANAGER Main Menu | Options
ALTER or CHANGE MANAGER Main Options Menu | JCLGEN options
JCL Generation Update Main Menu | Utility Options
Utility Options | BMC Utility Option Module Names

**Figure 77: JCL Generation BMC Utility Option Module Names Update panel**

```
--------- JCL GENERATION BMC UTILITY OPTION MODULE NAMES UPDATE ----------
COMMAND ===>
Type module names and press Enter.
Press PF3 or END to return to the main panel.

- COPY PLUS . . . Default ACP$OPTS
- REORG PLUS . . . Default ARU$OPTS
- LOADPLUS . . . Default AMU$OPTS
- UNLOAD PLUS . . . Default ADU$OPTS
- RECOVER PLUS . . . Default AFR$OPTS
- CHECK PLUS . . . Default ACK$OPTS
- DASD MANAGER PLUS . Default blank
```

1. Specify the name of the installation options module for the BMC utilities.
   
   If you use any of the BMC utilities, the installation options module names will be listed in the AEXIN input stream in the JCL.

2. Press END to save your changes, and to return to the ALTER or CHANGE MANAGER Main Menu.

---

### Setting the online reorg options

Use the Online Reorg Utility Options panel to specify the options for reorganizing table spaces.

**Note**

CATALOG MANAGER and DASD MANAGER PLUS currently do not use the values on the Online Reorg Utility Options panel.

---

### To set the JCL options for online reorg options

1. Use the following menu selections to display the Online Reorg Utility Options panel (Figure 78 on page 209):
<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>Utility Options</td>
</tr>
<tr>
<td>Utility Options</td>
<td>Online Reorg Options</td>
</tr>
</tbody>
</table>

Figure 78: Online Reorg Utility Options panel

```
COMMAND ===> ONLINE REORG UTILITY OPTIONS

BMCREORG XBMID: XBMB
REORG MAPPING TABLE: J234.JFLTBMAB
REORG MAPPING DATABASE: XXXX

NOTE: The mapping table full length cannot exceed 72 characters
NOTE: The mapping database full length cannot exceed 8 characters
```

2. In the **BMCREORG XBMID** field, specify the BMC EXTENDED BUFFER MANAGER (XBM) subsystem (SSID) that the REORG PLUS utility accesses.

REORG PLUS uses XBM or its XBM SNAPSHOT UPGRADE FEATURE (SUF) technology to create a snapshot of the data sets to be reorganized. ALTER and CHANGE MANAGER use this value when reorganizing a table space by using an online reorganization (SHRLEVEL CHANGE).

The value of the SSID can be from 1 to 8 characters long.

3. In the **REORG MAPPING TABLE** field, specify the name of the mapping table that the IBM REORG utility uses to map the row IDs (RIDs) in the source table to the RIDs in the target table.
If you want to use the system default values of IBM REORG, (not define a mapping table or a mapping database) for online reorg (SHRLEVEL CHANGE), perform the following steps:

1. On the Online Reorg Utility Options panel, do not specify a mapping database name.

2. On the Online Reorg Utility Options panel, do not specify a mapping table name.

3. On the ALTER and CHANGE MANAGER Analysis Options panel, set the **Require Mapping Table for IBM** field to **N**. See “Setting the Analysis options” on page 214.

   The name can be from 1 to 72 characters long, and can contain the **&ZUSER** or **&USERID** symbolic variable.

4. **(DB2 Version 11 and later)** In the **REORG MAPPING DATABASE** field, specify the name of the default database that the IBM REORG utility uses when it implicitly creates a mapping table.

   **Note**
   - On the ALTER and CHANGE MANAGER Analysis Options panel, the **Require Mapping Table for IBM** field must be set to **N**. See “Setting the Analysis options” on page 214.
   - The full length of the mapping database can be from 1 to 8 characters long.
   - You cannot define both the mapping database and the mapping table.

5. Press END to save your changes, and to return to the ALTER or CHANGE MANAGER Main Menu.
Setting the Execution worklist parallelism options

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the Execution Worklist Parallelism Options panel to specify the options for worklist parallelism in the CHANGE MANAGER component of the Database Administration solution.

**To set the JCL options for worklist parallelism**

1. Use the following menu selections to display the Execution Worklist Parallelism Options panel (Figure 79 on page 211):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN Options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>ALTER and CHANGE MANAGER Options</td>
</tr>
<tr>
<td>ALTER and CHANGE MANAGER Options</td>
<td>Worklist Parallelism Options (CHANGE MANAGER only)</td>
</tr>
</tbody>
</table>

**Figure 79: Execution Worklist Parallelism Options panel**

```
AJXOWPL ------- EXECUTION WORKLIST PARALLELISM OPTIONS UPDATE -----------------
COMMAND ===> 
Type data and press Enter. Press PF3 or END to return to the main panel.
Run Worklist in Parallel mode . . N (Y/N)
Start XIM . . . . . . . . . . . . N (Y/N)
Display Parallel Trace . . . . N (Y/N)
Minimum XIM Initiators . . . 2 (1 - 8)
Maximum XIM Initiators . . . 3 (1 - 32)
XIM Proc Name . . . . . . . . . XIMACM
XIM Group Name . . . . . . . . XIMACM
```

NOTE: The number of maximum initiators will determine the number of data set allocations.

JCL Generation inserts the keywords for the selected override options into the AEXPIN input stream in the JCL. These keywords override the default keywords in the POF. The Execution component uses the AEXPIN input stream to control parallel processing. Execution uses the AEXPIN input stream in conjunction with the AEXIN input stream.

Table 24 on page 212 lists the options on the panel, the AEXPIN keyword, and the AJXPOFIN keyword that is overridden by the AEXPIN keyword.
Table 24: Execution worklist parallelism options

<table>
<thead>
<tr>
<th>Option</th>
<th>AEXPIN keyword</th>
<th>AJXPOFIN keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run Worklist in Parallel mode</td>
<td>PARALLEL YES</td>
<td>ACM_PARALLEL_WORKLST</td>
</tr>
<tr>
<td>Start XIM</td>
<td>XIMSTART YES</td>
<td>ACM_PARALLEL_XIMSTRT</td>
</tr>
<tr>
<td>Display Parallel Trace</td>
<td>TRACE YES</td>
<td>ACM_PARALLEL_XIMTRCE</td>
</tr>
<tr>
<td>Minimum XIM Initiators</td>
<td>MAXINITS</td>
<td>ACM_PARALLEL_MAXINIT</td>
</tr>
<tr>
<td>Maximum XIM Initiators</td>
<td>MININITS</td>
<td>ACM_PARALLEL_MININIT</td>
</tr>
<tr>
<td>XIM Proc Name</td>
<td>XIMPROC</td>
<td>ACM_PARALLEL_XIMPROC</td>
</tr>
<tr>
<td>XIM Group Name</td>
<td>XIMGROUP</td>
<td>ACM_PARALLEL_XIMGRP</td>
</tr>
</tbody>
</table>

2 In the **Run Worklist in Parallel mode** field, type **Y** to enable worklist parallelism.

To run a worklist in parallel, the PARALLEL keyword must also be specified in the ALUIN input stream. For more information, see the *ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2*.

3 In the **Start XIM** field, type **Y** to start XIM automatically when you execute a worklist.

XIM is started only on the OS/390 or z/OS image on which you submit a job. If you use a data sharing environment, XIM is not started on other images.

4 In the **Display Parallel Trace** field, type **Y** to display trace messages in an output log during the execution of the worklist.

5 In the **Minimum XIM Initiators** field, type the minimum number of XIM initiators.

6 In the **Maximum XIM Initiators** field, type the maximum number of XIM initiators.

   **Note**

   If the maximum number of XIM initiators that you specify is larger than the number of objects in the scope of the worklist, your DASD might not be used efficiently.

7 In the **XIM Proc Name** field, type the name of the started task procedure for XIM.
**Note**

BMC recommends that the name of the started task procedure for XIM be unique for each instance of XIM that is running on an OS/390 or z/OS image. In addition, you should copy the procedure into a system PROCLIB data set. BMC also recommends that you use the suggested default value unless you have a specific requirement for another value.

8. In the **XIM Group Name** field, type the group name for XIM.

**Note**

The group name for XIM must be unique for each instance of XIM that is running on an OS/390 or z/OS image. BMC recommends that you use the suggested default value unless you have a specific requirement for another value.

9. Press END to save your changes, and to return to the **CHANGE MANAGER Main Menu**.

### Setting the worklist options

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the ALTER or CHANGE MANAGER Worklist Options panel to specify the options for sorting a worklist.

**To set the JCL options for worklist sorting**

1. Use the following menu selections to display the ALTER and CHANGE MANAGER Worklist Options panel (Figure 80 on page 214):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN Options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>ALTER and CHANGE MANAGER Options</td>
</tr>
</tbody>
</table>
From this menu | Select this item and press Enter
---|---
ALTER and CHANGE MANAGER Options | Worklist Options

**Figure 80: ALTER and CHANGE MANAGER Worklist Options panel**

AJXOWKL ---------- ALTER and CHANGE MANAGER WORKLIST OPTIONS ------------------
COMMAND ===> 
Type data and press Enter. Press PF3 or END to return to the main panel.
Sort Worklist . . . . . . . . (Blank - No Sorting
A - Automatic sorting
C - Object Cardinality sorting
N - Table Name sorting)
Print Sort Worklist Messages Y (Y - Print Messages,N - No messages)

2 In the **Sort Worklist** field, type A, C, or N (or leave the field blank) to specify the sorting method, as shown in **Table 25 on page 214**:  

**Table 25: Worklist sorting options**

<table>
<thead>
<tr>
<th>For this method</th>
<th>Type</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sorting the worklist by each table’s cardinality in a descending sequence</td>
<td>C</td>
<td>None</td>
</tr>
<tr>
<td>Sorting the worklist by table order in an ascending sequence, according to the table owner and table name</td>
<td>N</td>
<td>None</td>
</tr>
<tr>
<td>Sorting the worklist by either table cardinality or table order, depending on whether the worklist is processed in parallel</td>
<td>A</td>
<td>If the Database Administration solution is used to process the worklist in parallel, Analysis sorts the worklist by table cardinality. Otherwise, it sorts the worklist by table order.</td>
</tr>
<tr>
<td>Generating the objects in the worklist in an unsorted, random order</td>
<td>None</td>
<td></td>
</tr>
</tbody>
</table>

3 In the **Print Sort Worklist Messages** field, type Y to record in the SYSPRINT data set and in the worklist the amount of time to sort a worklist.

4 Press END to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.

**Setting the Analysis options**

Use the ALTER and CHANGE MANAGER Analysis Options panel to specify the options for dynamic sortwork data sets, the mapping table name, and the default output class.
To set the Analysis options

1. Use the following menu selections to display the ALTER and CHANGE MANAGER Analysis Options panel (Figure 81 on page 215):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN Options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>ALTER and CHANGE MANAGER Options</td>
</tr>
<tr>
<td>ALTER and CHANGE MANAGER Options</td>
<td>Analysis Options</td>
</tr>
</tbody>
</table>

Figure 81: ALTER and CHANGE MANAGER Analysis Options panel

```
------------------ ALTER and CHANGE MANAGER ANALYSIS OPTIONS ------------------
COMMAND ===> 
Type data and press Enter. Press PF3 or END to return to the main panel.
Dynamic Sortwork Number . . . . .  32 (1 - 255)
Dynamic Sortwork Unit . . . . . . .  SYSDA (SYSDA, 3380, etc)
Require Mapping Table for IBM Reorg . Y (Y/N)
Sysout class . . . . . . . . . . . .  A
```

2. In the Dynamic Sortwork Number field, type the number of dynamically allocated sortwork data sets that the REORG PLUS or IBM REORG utility uses.

3. In the Dynamic Sortwork Unit field, type the unit for dynamically allocated sortwork data sets.

   The value of the unit can be a name from 1 to 8 characters long, blank, or NONE. To omit the UNIT parameter from the JCL, specify NONE.

4. In the Require Mapping Table for IBM Reorg field, type Y to include the name of the mapping table in the syntax for the IBM REORG utility.

   The IBM REORG utility uses the mapping table to map the row IDs (RIDs) in the source table the RIDs in the target table. For more details, see “Setting the online reorg options” on page 208.

   **Note**
   The REORG PLUS utility invokes the IBM DSNUTILB utility control program to enable certain features. If you have specified to use the REORG PLUS utility, you still need to specify mapping table information. For information about the features for which REORG PLUS invokes DSNUTILB, see the REORG PLUS for DB2 Reference Manual.

5. In the Sysout class field, type the name of the default output class.
Setting user variables

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the User Defined Variables panel to specify character variables. Each variable has a corresponding symbolic variable that you can use in job cards or data set prefixes.

To set the JCL options for user-defined variables

1. Use the following menu selections to display the User Defined Variables panel (Figure 82 on page 216):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>User Defined Variable Values</td>
</tr>
</tbody>
</table>

**Figure 82: User Defined Variables panel**

```
COMMAND ===> USER DEFINED VARIABLES UPDATE
```

Type data and press Enter.

Character Variables:
- User variable 1 ..... (Symbolic &UVR1)
- User variable 2 ..... (Symbolic &UVR2)
- User variable 3 ..... (Symbolic &UVR3)
- User variable 4 ..... (Symbolic &UVR4)
- User variable 5 ..... (Symbolic &UVR5)

NOTE: Symbolic variables cannot be input to these values

2. Specify the values for the variables.

   The maximum length of a variable name is eight characters.

3. Press END to save your changes, and to return to the CHANGE MANAGER Main Menu.
Creating a user POF

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the Product Options File (POF) Functions panel to create a user POF or update the values in your ISPF profile. The panel displays the data set name of the initial POF. The panel also displays the value of the POFDATE parameter in the initial POF that was last used to update the ISPF profile.

To create a user POF

1 Use the following menu selections to display the JCL Generation Product Options File (POF) Functions panel (Figure 83 on page 217):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>Product Option File (POF) Functions</td>
</tr>
</tbody>
</table>

**Figure 83: JCL Generation Product Options File (POF) Functions panel**

```
COMMAND ===> JCL GENERATION PRODUCT OPTIONS FILE (POF) FUNCTIONS

Initial POF name: AUS.DOPSEC.CNTL(JX10QBDC)
Initial POF name Different from previous? N
BROWSE Initial POF . . . . . . . . . . N (Y/N)
MSGCLASS for POF Diagnostic Messages . X
POFDATE value used for initial POF Refresh Compare: 2011/02/09 08:15:23
Type User POF Name below: .. AUS.DOPSEC.CNTL(JX10QBAC)
BROWSE, EDIT, VALIDATE User POF . . . . . . . . N (B/E/V/N)
RESET Profile Variables from User POF that are marked with Refresh ",(R)" . . . . . . N (Y/N)
RESET All Profile Variables from User POF . . . . . N (Y/N)
WRITE User POF data set from Profile Variables . . N (Y/N)
-- Edit/Validate will update the User POFDATE --
```

2 In the **Type User POF Name below** field, replace the displayed name (the initial POF) with the name of the data set for a user POF.

The name can be either an existing sequential, 80-column data set or a member of a PDS.

**Note**

Consider creating user POF to set the options for processing a worklist in parallel.
3 In the **WRITE User POF data set from Profile Variables** field, type **Y** to write the ISPF variable values (located in the ISPF profile) to the user POF.

4 Press **Enter**.

## Updating a user POF

You can update a user POF by using the options panels or by directly editing the file.

### To update the user POF in the JCL Generation options panels

1 In the various options panels, specify your changes to the JCL Generation options.

2 Use the following menu selections to display the JCL Generation Product Options File (POF) Functions panel:

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>Product Option File (POF) Functions</td>
</tr>
</tbody>
</table>

3 In the **Type User POF Name below** field, type the name of the data set for your user POF.

4 In the **WRITE User POF data set from Profile Variables** field, type **Y**.

5 Press **Enter**.

### To update the user POF directly

1 From the JCL Generation Product Options File (POF) Functions panel, in the **Type User POF Name below** field, type the name of the data set for your user POF.

2 In the **BROWSE, EDIT, VALIDATE User POF field**, type **E**.

   **Note**

   You can edit the user POF or the initial POF by using the ISPF edit macro AJXPODAT from the `HLQ.BMCCLIB` library. This library must be in your SYSPROC concatenation.

3 Edit and save the file.

4 In the **RESET All Profile Variables from User POF** field, type **Y** to update all of the ISPF variables in the ISPF profile with the variables in the user POF.
**Note**
You can update all of the ISPF variables in the ISPF profile with the variables in the POF by using one of the following **Command** line commands:

- To use the variables in the initial POF, enter **TSO POFRESET**.

- To use the variables in a specified initial or user POF, enter **TSO POFRESET POF (dataSetName(POFMember))**.

To enable these commands, you must modify the POFRESET CLIST. The CLIST is located in the *HLQ.BMCCLIB* library. A current copy of this CLIST must be in the same SYSPROC concatenated library as your other CLISTs.

5 Press **Enter**.

**Using multiple POFs**

If you want to use different values for different applications, consider using more than one user POF. When you use multiple POFs, you can reset the values from a user POF that contains specifications for a particular application.

Some sample scenarios follow:

**Scenario 1**

The Payroll department needs backup copies of their data sets on tape, but other departments do not need backup copies.

You can create one user POF for the Payroll department and one for the other departments. Whenever you work with payroll objects, you can specify your payroll POF name to reset the profile variables from that POF.

**Scenario 2**

Your company is a service provider for several customers.

By using a separate POF for each customer, you can accommodate each customer’s naming standards.

**Scenario 3**

You have different requirements for test and production data.
For test data, you want to store the copies on DASD. For production data, you want to store the copies on tape. You can use separate POFs for test data and production data.

1. Follow the steps in “Creating a user POF” on page 217 to create an additional POF.

2. On the Command line, enter `TSO POFRESET POF (dataSetName (POFMember)).`

   **Note**
   To specify the use of a different user POF, from the JCL Generation Product Options File (POF) Functions panel, in the **Type User POF Name below** field, type the name of the data set for your customized POF.

---

**Refreshing the initial POF**

Assume that your shop has revised its standards for naming data sets.

To enforce the new standards, you need to refresh the initial POF so that users will get the updated values.

The JCL Generation component uses the value of the POFDATE keyword and the refresh attribute when determining whether to reset the ISPF profile variables to the updated POF values. The refresh attribute of a POF keyword value indicates that the ISPF profile variable should be reinitialized from the POF value if one of the following conditions exists:

- The value of the POFDATE keyword is greater than that saved in the ISPF profile.
- The name of a new initial POF is different from the name of the POF that is saved in the ISPF profile.

**To refresh the initial POF**

1. Edit the initial POF outside of the product.

2. Change the value of the POFDATE keyword to the current date.

3. Append the refresh attribute `(R)` to the values that you want to update.

4. Save the POF.

After you save the changes, users will receive the updated ISPF variables the next time they invoke one of the products. However, the users can still use their existing user POFs, which might not contain the updated values. To ensure that
the user POFs use the updated values, tell the users to reset all of their profile variables from the revised initial POF. If the users typically reset all of their variables from their user POFs, they must ensure that the updated values are included.

Alternatively, you can specify the name of a new initial POF in the POFDS installation option.

## Generating POF reports

Periodically, you might need to review POF keyword values, determine whether values are missing, or diagnose a problem.

The following reports can assist you:

- **The POF Validation Report** lists a POF keyword, the action taken on the keyword, and the value of the keyword in the POF. For example, you can generate this report when you create a new user POF and change the values of several keywords. The report shows the changes, the number of values that were refreshed, and any errors that resulted.

- **The Variables Initialized with Default** report lists the keywords that are missing from the initial POF and the default ISPF variables that are used to populate the keywords. You can generate this report when you want to view the new keywords and their values for a release.

### To generate the reports

1. From the JCL Generation Product Options File (POF) Functions panel, in the **MSGCLASS for POF Diagnostic Messages** field, type the MSGCLASS for the SYSOUT field that is used to display messages.

   Consider using a SYSOUT class that is designated to go to the held queue so that you can view the output. Two SYSOUT files are allocated: AJXPOFER and AJXPOFVL.

   **Note**

   The default value for the **MSGCLASS for POF Diagnostic Messages** field is blank, which indicates that JCL Generation does not generate a report when you invoke the product.

   When you invoke the product or reset the POF, the reports are listed on the output for your TSO session.
Reusing a POF in a subsequent installation

Assume that you customized the values in your POF, and now you are installing a new release of a product.

To avoid having to customize the values again, you can specify that the Installation System use your existing POF to populate the values in the new initial POF. The new POF will contain your current values plus any new keywords (and their values) for the new release.

To reuse a POF

1. Run the Installation System.

2. From the Install System JCL Generation File Information panel, in the **Use Existing POF to Populate the New Product Options File** field, type **Y**.

   ![Figure 84: Reusing an existing POF](image)

   **AJXP041 Install System DB2 Administrative Components**

   **Command ===>*

   **Product Options File name (POF)**  . . . . . . . . . . ZAJXBPOF

   **Use Existing POF to Populate the New Product Options File**. **Y** (Y/N)

   **Press PF1 HELP for more information regarding the POF, its use, and how to avoid having to reenter the information for SSID installs.**

   **Verify the ACS Common SQL API Synonym Qualifier and the Collection ID. BMC Software Inc. recommends accepting the defaults. To make changes, type over the values shown.**

   **Qualifier of Product Synonyms**  . . ZCS111S1

   **ACS Collection ID**  . . . . . . . ZCS111_D_MAIN

   **Execution for DB2 Collection ID**  . . . . . . . ZEX111_D_MAIN

   **Press Enter to continue or F12 to go back.**

3. Enter the names of the data set and member for the existing POF (Figure 84 on page 222).

   For the member name, use the name of the POF that is used as the initial POF when you invoke the product.

   ![Figure 85: Specifying the name of the existing POF](image)
Overriding POF values in SLIBs

One of the primary advantages of using POFs is that you can customize your JCL without having to modify your SLIBs.

Nonetheless, you might need to modify your SLIBs from time to time. SLIB variables (or ISPF variables) are used in the SLIBs. Some of these SLIB variables correspond to the parameters in the POF. Note, however, that the names of the SLIB variables differ from the names of the POF keywords.

For example, if you specify the data set prefix for local primary copies (Figure 86 on page 223), the name of the primary copy data set resolves to the following name without modifications to the SLIB:

<SSIDname>.IC.T.ICPY.<databaseName>.<tableSpaceName>.<ddname>

For data sets that are not dynamically allocated, JCL Generation appends the ddname to the prefix to create the name of the data set.

**Figure 86: Specifying the prefix for a copy data set**

Now, assume that your site’s DBA decides that users should not have the ability to change the data set prefix from the options panels. The DBA can override the value in the SLIB. By specifying the value for the copy data set in the AJX#DSNS SLIB (Figure 87 on page 223), the DBA can uphold your site’s naming standards.

**Figure 87: Changing the SLIB variable for the copy data set in AJX#DSNS**
After changing an SLIB variable, the DBA should use JCL Generation to test the changes. If the SLIB is coded correctly, the DBA must then recompile the SLIB. The DBA can use the SLIB compiler tool that is supplied with the Administrative products to compile the SLIB.

For more information about testing the changes or using the SLIB compiler, see *ALTER and CHANGE MANAGER for DB2 User Guide*, Volume 2.

### Creating OUTPUT or TEMPLATE descriptors for copies

In ALTER and CHANGE MANAGER, you can use BMC and IBM utilities to dynamically allocate data sets that are used to create valid image copies.

You can use an option in the POF to create an OUTPUT or a TEMPLATE descriptor for the Analysis component of ALTER and CHANGE MANAGER. When generating a worklist, Analysis uses that descriptor for local and recovery primary and backup copies (Figure 88 on page 224).

**Figure 88: OUTPUT descriptor in worklist**

```
=BMCC 001250
  OPTIONS IXDSNUM ALL
  OUTPUT DCPYLOCP
  UNIT SYSDA
  DSNAM= PAYROLL.IC.LP.&DB.&TS
  MODELDCB SYS1.MODEL
  COPY INDEX ACMX01.Z_X06SSB1
  COPYDDN (DCPYLOCP)
```

**To create OUTPUT or TEMPLATE descriptors for copies**

1. From the ALTER and CHANGE MANAGER Main Menu, select *Options* and press Enter.

2. Select JCLGEN Options and press Enter.
3 Select Individual Data Set Options and press Enter.

4 Select Copy and press Enter.

5 Select Local Primary Copy and press Enter.

6 Type the name of the prefix for the data set that will be dynamically allocated (Figure 89 on page 225).

**Figure 89: Specifying the prefix for a dynamically allocated data set**

```
Enter Data Set Prefix below:

. . PAYROLL.IC.LP..&DB..&TS
Unit Name . . . . . . . . . . . SYSDA (SYSDA, TAPE, etc)
Primary Space . . . . . . . . . 10 (Cylinders)
Secondary Space . . . . . . . . . 2 (Cylinders)
Tape EXPDT. . . . . . . . . . . (Blank or YYDDD or YYYY/DDD)
Tape RETPD. . . . . . . . . . . (Blank or 0 - 9999 days)
SMS Data Class. . . . . . . . . (Blank or Data Class)
SMS Storage Class . . . . . . . (Blank or Storage Class)
SMS Management Class . . . . (Blank or Management Class)
Threshold Value . . . . . . . . 0 (Cylinders, 0 means no Threshold)
Alternate Unit Name . . . . . (SYSDA, TAPE, etc)
Alternate SMS Data Class. . . (Blank or Data Class Name)
Alternate SMS Storage Class . . (Blank or Storage Class Name)
Alternate SMS Management Class (Blank or Management Class Name)
```

If you are using foreground Analysis to generate the worklist, JCL Generation uses the prefix in your ISPF profile variables to create the OUTPUT descriptor. However, if you are using batch Analysis, JCL Generation generates a partial POF from your ISPF profile variables and includes this partial POF in the batch Analysis JCL (Figure 90 on page 225).

**Figure 90: Excerpt of batch Analysis JCL with POF keywords**

```
//AJXPOFIN DD   *
...
PCPY1_DATACLASS_ALT=
PCPY1_DATACLASS=
PCPY1_EXPDT=
PCPY1_MGMTCLASS_ALT=
PCPY1_MGMTCLASS=
PCPY1_PREFIX=PAYROLL.IC.LP..&DB..&TS
PCPY1_PRIQTY=10
PCPY1_RETPD=
PCPY1_SECQTY=2
PCPY1_STACK=N
PCPY1_StORCLASS_ALT=
```
Glossary

A

ACM

The product code that BMC uses to identify the CHANGE MANAGER product.

ACT

The product code that BMC uses to identify the CATALOG MANAGER product.

action code

A one-character or two-character command that you can enter on one of the lines of a list panel. Although you can only enter one command per line, you can enter multiple commands on a single panel.

Administrative Products for DB2

A collection of products from BMC that includes ALTER, CATALOG MANAGER, CHANGE MANAGER, and DASD MANAGER PLUS. These integrated products are designed to help database administrators, system programmers, and application developers automate the tasks associated with the implementation and administration of a DB2 Universal Database for z/OS system.

When all Administrative products are installed properly, they can access some of each other’s functionality. In addition, these products can access the IBM and BMC utilities.

AEXIN

The ddname of the input stream that the Execution component uses.
AEXPIN

The ddname of the input stream that the Execution component uses to control parallel processing.

AEXPRINT

The ddname of diagnostic output of the Execution component. This diagnostic output data set contains all output from the Execution process, including DB2 for z/OS utility messages, BMC utility messages, dynamic SQL messages, IDCAMS messages, and any other messages that are generated by the actions of the worklist. AEXPRINT is frequently referred to as the worklist execution log.

AEXPRnnn

The ddname of one of the initiator outputs of the Execution component. When the worklist parallelism feature of the Database Administration solution is used, this output data set contains AEXPRINT output from each BMC Cross-System Image Manager (XIM) initiator.

AEXPTRAC

The ddname of the trace output of the Execution component. When the worklist parallelism feature of the Database Administration solution is used, this output data set contains tracing records.

AEXSYnnn

The ddname of one of the initiator outputs of the Execution component. When the worklist parallelism feature of the Database Administration solution is used, this output data set contains system messages and job information for each BMC Cross-System Image Manager (XIM) initiator.

AJX variables

A group of variables (JOB, STEP, and DD) that are used during Job Control Language (JCL) generation. JOB global variables are set once per JCL creation session. STEP global variables are set at the beginning of a JCL creation session and are updated when a new JOBSTEP is detected. Data Definitions (DD) variables are local variables that are set for each creation of a JCL DD entry. Descriptions of these variables are provided in HLQ.BMCSLIB($AJXDOCV).

AJXIN

The ddname of the input stream that the Batch Execution JCL Generation component uses.
AJXPOFIN

The ddname of the override input stream that the Batch Execution JCL Generation component uses.

AJXPOFVL

The ddname of the product options file (POF) validation report that the Batch Execution JCL Generation component uses.

AJXPRINT

The ddname of the diagnostic output of the Batch Execution JCL Generation component.

alloc unit

The allocation unit that is used for space estimation calculations, volume placement, and primary and secondary quantities. Possible values are K (kilobytes), T (tracks), or C (cylinders). The default value comes from the installation options modules.

ALTER for DB2

A BMC product that provides advanced database administration and manipulation within a single DB2 subsystem. ALTER streamlines the process of changing and migrating database objects, handles analysis for both changes and migrations, and automatically generates SQL, DB2 commands, and utilities.

alter-type work ID

A work ID that ALTER and CHANGE MANAGER use to perform modifications to the local DB2 subsystem.

ALU

The product code that BMC uses to identify the ALTER product.

ALUIN

The ddname of the input stream that the Import, Baseline, Baseline Report, Compare, Analysis, and CM/PILOT components of CHANGE MANAGER use.
ALUPRINT

The ddname of the diagnostic output of the Import, Baseline, Baseline Report, Compare, and Analysis components of CHANGE MANAGER.

Analysis

A component of ALTER and CHANGE MANAGER that reads the change and migration requests for a particular work ID and generates a worklist to implement the requests. Analysis checks for consistency with the DB2 catalog before it builds the worklist.

Application

In the CM/PILOT component, the association of a group of CHANGE MANAGER profiles that are used repeatedly for the change management tasks of a specific DB2 application.

ASU

The product code that BMC uses to identify the DASD MANAGER PLUS product.

attribute

A value that defines certain properties of an object. Each attribute of an object can occur only once and has a single value from a finite list of possible values. For example, some of the attributes of the TABLE object are database, table space, and EDITPROC. Attributes differ from sub-elements because sub-elements can occur multiple times and can have attributes of their own. For example, columns are sub-elements of tables.

AUTHID

See “authorization ID” on page 230.

authorization ID

An identifier that is allowed a set of privileges. An example of the authorization ID is the owner of a table space, database, storage group, or synonym. An authorization ID is the implicit qualifier of a table, view, alias, or index name.

auxiliary index
An index on an auxiliary table. Each index refers to a large object (LOB) column.

**auxiliary list**

A list of several auxiliary objects (such as auxiliary table spaces, tables, or indexes) of only one type.

**auxiliary table**

A table that contains a single large object (LOB) column. An auxiliary table resides in an auxiliary (or LOB) table space.

**auxiliary table space**

A nonpartitioned table space that contains the data for a large object (LOB) column in an auxiliary table.

**B**

**base table**

A table that contains a ROWID column and the definition for a large object (LOB) column. A base table is incomplete if it does not contain any auxiliary objects, such as auxiliary table spaces, tables, and indexes.

**base table space**

A table space that contains base tables.

**baseline**

A component of CHANGE MANAGER that captures a set of DB2 structure definitions from either the DB2 catalog or a DDL file at a specific point in time.

**baseline name template**

A template that a baseline profile can contain. This template is used to create the names of the baselines that are established with the profile. The template can include arbitrary text and the special character sequences #### and @@@@@ or @@@@@@@, or a combination of both types of sequences. When the baseline is created, ascending numbers replace the ####

---

**Glossary**

231
sequence and the current date replaces the @@@@@@@@ or @@@@@@@@@@@ sequence. For @@@@@@@@, the date is in the form YYMMDD. For @@@@@@@@@@@, the date is in the form YYYYMMDD.

baseline profile

A BMC object that contains the information that is necessary to establish a baseline. The two types of baseline profiles are catalog and DDL.

batch component

A component that can run in batch mode. Batch components of CHANGE MANAGER include Analysis, Baseline, Baseline Report, Compare, Execution, and Import.

BMC object

A logical entity that contains data that is necessary for performing tasks. BMC objects have a two-part name (a,b). The BMC objects for CHANGE MANAGER include work IDs, worklists, unload data sets, baselines, profiles, CDL files, DDL files, internal tables, task IDs, CM/PILOT worklists, applications, and script tables.

BMCCOPY

The BMC COPY PLUS utility that is used to create an image copy. The short form of the command is BMCI.

BMCLOAD

The BMC LOADPLUS utility that is used to load DB2 tables. The short form of the command is BMCL.

BMCREORG

The BMC REORG PLUS utility that is used to perform a reorganization. The short form of the command is BMCR.

BMCSTATS

A command that invokes the statistics collecting function of the DASD MANAGER PLUS product. The short form of the command is BMCS. BMCSTATS is similar to the IBM RUNSTATS utility. It provides the same statistics plus many additional statistics.
BMCUNLOAD

The BMC UNLOAD PLUS utility that is used to unload data from a full image copy of one or more tables in a table space. The short form of the command is BMCD.

CANCEL (CAN)

An ISPF command that returns you to the previous panel without saving any changes made to the current panel.

catalog indirection

An optional method of implementing the Administrative products that allows them to access the DB2 catalog indirectly when making information queries. Catalog indirection is accomplished by using aliases that point to a copy of the DB2 catalog. The major benefit is to reduce catalog contention.

catalog baseline

A baseline that is established on part of the DB2 catalog. A catalog baseline must include a scope, which might be defined directly in a baseline profile or by reference to a scope in a migrate profile.

CATALOG MANAGER for DB2

A tool that is designed to automate the day-to-day tasks associated with administering DB2. This product provides an interactive, intuitive, and easy-to-use interface for submitting DB2 commands and retrieving catalog information using qualified lists, wildcard searches, and dependency lists. CATALOG MANAGER provides the ability to create and drop DB2 objects, re-create dropped structures and data, and browse and edit table data.

CDL

See “Change Definition Language (CDL)” on page 233.

Change Definition Language (CDL)

A BMC proprietary language that is used to specify changes to DB2 data structures.
CHANGE MANAGER for DB2

A BMC product that enables database administrators, system administrators, and developers to manage user applications and individual database objects globally. As a robust extension of the ALTER product, CHANGE MANAGER automates data structure changes across multiple DB2 subsystems and between DB2 and data modeling tools by providing a way to implement, migrate, synchronize, and back out data structure changes while preserving structure modifications that might be unique to a specific subsystem.

change rule

An element of a migrate-type work ID or a migrate profile. Change rules define the automatic object attribute changes that are made during a migration or change migration process. These rules can include changes, or sub-element inclusion or exclusion. Change rules can also force VCAT definitions for table spaces and indexes.

checkpoint

A point at which information about the status of a job and the system can be recorded so that the job step can later be restarted.

See also "sync point" on page 252.

child

A DB2 object that contains the foreign keys which reference the primary key in a parent table.

See also "parent" on page 247.

CLIST

Command list.

CM/PILOT for DB2

A component of CHANGE MANAGER that automates the DB2 change management processes that you perform using CHANGE MANAGER.

CM/PILOT worklist

A data set that contains the ordered commands, keywords, and parameters that CHANGE MANAGER needs to process a task ID.
command

A token that you can enter at the command prompt on a panel.

*See also* “action code” on page 227.

commit

An operation that terminates a unit of work. A commit releases all locks. Data that was changed is now consistent.

Compare

A component of CHANGE MANAGER that identifies the differences between two sets of data structures and then generates a CDL file. You can review the differences and decide how to apply the differences to implement the necessary changes. You can compare data structures that are stored in a DDL file, baseline, worklist, or DB2 catalog.

Compare1

The primary input to the compare process.

*See also* “source” on page 250.

Compare2

The secondary input to the compare process.

*See also* “target” on page 252.

cOMPONENT

A major functional unit of ALTER or CHANGE MANAGER, such as Analysis, Execution, Specification, or Import.

cONSTRAINT

*See* “referential constraint” on page 249.
Cross-System Image Manager (XIM)

A BMC technology that provides sysplex performance improvements by enabling the distribution and management of discrete units of work (UOW) across one or more OS/390 and z/OS systems. The BMC products that use XIM can divide single, long-running tasks into multiple parallel tasks that can be run across multiple computers in the sysplex, thus decreasing the overall elapsed time. The products can also be used with XIM in a data sharing environment on a single OS/390 or z/OS image.

D

DASD MANAGER PLUS for DB2

A BMC product that automates utility generation, gathers comprehensive statistics, monitors changes in the database, and enables you to perform maintenance based on the condition of the data instead of a rigid schedule.

Data Control Language (DCL)

A category of SQL statements that control data security.

data definition language (DDL)

A category of SQL statements that create, modify, or delete database objects.

data definition name (ddname)

The name of a data definition (DD) statement in job control language (JCL) that corresponds to a data control block that contains the same name.

Data Manipulation Language (DML)

In the CM/PILOT component of CHANGE MANAGER, SQL-like statements that can be used to update, delete, and migrate data structures.

data set sizing

The process of determining data set allocations, especially as used by the JCL Generation component. Data set sizing is distinct from space estimation.
data structure

An object that is defined in the DB2 catalog. Objects include storage groups, databases, table spaces, tables, indexes, foreign keys, views, synonyms, aliases, and triggers.

database administrator (DBA)

An individual who is responsible for the design, development, operation, security, maintenance, and use of databases.

database request module (DBRM)

A module that contains SQL statements which the DB2 precompiler has extracted from a source program.

DB2 catalog

System tables, maintained and used by DB2, that contain descriptions of DB2 objects such as tables, views, and indexes.

DB2 command

An instruction to the DB2 subsystem. Some example command processes enable you to start or stop DB2, display information on current users, start or stop databases, and display information about databases. DB2 commands always begin with a hyphen (-).

DBA

See “database administrator (DBA)” on page 237.

DBCS

See “double-byte character set (DBCS)” on page 239.

DD statement

Data Definition statement.

DDL
See “data definition language (DDL)” on page 236.

**DDL baseline**

A baseline that is established on a file that contains DDL.

**ddname**

See “data definition name (ddname)” on page 236.

**default options module (DOPTS)**

See “installation options module” on page 242.

**default value**

A predetermined value, attribute, or option that is assumed when no other is explicitly specified.

**delimited identifier**

An SQL identifier that is enclosed within escape characters.

**dependencies**

The name or values of objects which another object uses as part of its definition or as a hierarchical subordinate.

**dependent**

A child object (row or table) that has at least one parent.

**dependent object**

An object whose definition relies on the name or the values of another object. The dependent object references the other object.

**destination**
The intended receiving location for CDL or a worklist.

device type

The type of disk device used for DB2 data set allocation, such as 3380 and 3390, or generic types TAPE and CART.

DOPTS

See “installation options module” on page 242.

double-byte character set (DBCS)

A delimited set of characters in which each character is represented by two bytes. Katakana and other lowercase characters are nonstandard characters and must be contained within double quotes.

duplicate

An action that you can make on a database object. You can duplicate single or multiple objects within the same database system. If you want to create an object that is like an existing object, you can use the Create Like command to duplicate the existing object. You can then change the object name and make any other necessary changes.

E

edit procedure

See “EDITPROC” on page 239.

EDITPROC

An edit procedure that defines an editing routine to be invoked just after a record that corresponds to a table row is retrieved and just before that record is stored. Editing routines allow for data compression, decompression, and encrypting.
An ISPF command, similar to Enter, that validates and processes the information on a panel and returns you to the previous panel, but does not execute commands. This command is typically programmed on your keyboard as a function key, such as PF3.

**Enter key**

The key that executes any commands that have been specified. For a sequence of panels, the Enter key displays the next panel.

**Execution**

A component of ALTER and CHANGE MANAGER that carries out the commands in a worklist.

**exit routine**

A program (BMC, IBM, or user-written) that receives control from DB2 to perform specific functions. Exit routines run as extensions of DB2 (for example, authorization checking).

**F**

**field procedure**

See “FIELDPROC” on page 240.

**FIELDPROC**

A user-written exit routine that is designed to receive a single value and transform (encode or decode) it in any way that the user specifies.

**foreground component**

A component that can run in foreground mode. Foreground components in CHANGE MANAGER include Front End, Specification, Analysis, Compare, JCL Generation, and Import.

**Front End**

A component of ALTER and CHANGE MANAGER that acts as the interface between the user and the other components. Front End is an interactive ISPF dialog that is responsible for creating and maintaining BMC objects and facilitating the generation of JCL.
full-recovery baseline

A baseline that captures data and the data structure definitions at a specific point in time.

G

GDG

See “generation data group (GDG)” on page 241.

generation data group (GDG)

A finite number of data sets that are kept in chronological order. Each data set is a generation data set.

Group ID

In an outbound migrate profile, a four-character identifier that links locations (or application instances) together. If a migrate profile defines one or more Group IDs, Analysis and Compare generate a single output file (worklist or CDL) for each group. If groups are not defined, Analysis and Compare generate one output file for each location. Locations within a group must reside on the same physical DB2 subsystem.

H

hash value

A number that appears at the end of commands in worklists or CDL files. The hash value is generated based on the contents of the command line and allows the products to determine whether the line has been manually modified since the file was generated.

HLQ

High-level qualifier of a data set.
image copy

An exact reproduction of all or part of the data in a table space. IBM provides utility programs to make full image copies (copy the entire table space) or incremental image copies (copy only the pages that have been modified since the last image copy). The BMC COPY PLUS utility can perform the same function. You can make an image copy of an index.

Import

A component of CHANGE MANAGER that converts statements that are stored in a CDL, DDL, or DML file into change requests in a work ID. In ALTER, Import converts statements that are stored in a DDL file.

import or importation

The process of obtaining an object or an object-set definition from a file or an external database and applying it to an alter-type work ID.

inbound migrate profile

A profile that can be used with the Import component to automate changes to object attributes. An inbound migrate profile can contain only change rules (no locations or scope).

incremental DDL

The DDL that changes the data structures that exist in the DB2 catalog, by using either an alter strategy or a drop-then-rebuild strategy.

installation options module

An assembler module that contains keywords whose global values determine the operating environment for a BMC product.

internal table

A table that ALTER or CHANGE MANAGER use to store information.
ISPF skeletons

Data definition statement templates that JCL Generation uses. The skeletons are described in HLQ.BMCSLIB($AJXDOC).

J

JCL

Job Control Language that is used to execute processes in batch mode.

JCL DSN

The name of the data set that contains job control language (JCL). The data set must exist and can be partitioned or sequential. You must specify a member name for partitioned data sets. You can use symbolic variables.

JCL Generation (JCLGEN)

A component of ALTER and CHANGE MANAGER that constructs a job control language (JCL) file for running the components in batch mode. When you choose to build JCL, JCLGEN is passed to the worklist that contains the control statements. ALTER and CHANGE MANAGER resolve all data set names that are entered with symbolic variables on the interface panels. JCLGEN resolves all data sets that are passed from the option panels and the unload data sets that are used by the Execution facility. The generated JCL includes data definition statements (ddnames) for all data sets that are needed by Execution, as well as the EXEC statement for the program and any necessary control parameters.

JCL variable display

The resulting output of a user option that includes debugging comments within any generated JCL. All AJX-prefixed variables are displayed as //* comments in the JCL to assist in diagnosing JCL Generation problems.

JCLGEN

See “JCL Generation (JCLGEN)” on page 243.

job

A batch unit of work that is defined by JCL, a work ID, and a worklist to perform tasks. ALTER and CHANGE MANAGER use the information that you supply for a work ID to generate the
worklist, which provides the BMC JCL Generation component (JCLGEN) with the information that is necessary to build the JCL to run a job.

L

large object (LOB) column

A type of column that is used to store large objects (LOBs), such as images, audio, video, text, or graphics, as strings. The data type of the column is defined as LOB (such as a binary large object, or BLOB; character large object, or CLOB; or double-byte character large object, or DBCLOB) or as a distinct or user-defined type (UDT) that is based on a LOB data type. A LOB column resides in an auxiliary table.

large object (LOB) table space

See “auxiliary table space” on page 231.

LINK library

A partitioned, cataloged data set that is used to store and retrieve all or part of a program in a form that is suitable for loading into main storage for execution. The LINK library contains executable modules that perform a product’s processes.

LOB DATA MOVER

In the Database Administration solution, a program that is used to unload and load data that is contained in large object (LOB) columns.

location

An arbitrary identifier that groups change rules for a particular destination. An outbound migrate profile can define one or more locations. A location is not a DB2 location identifier.

M

menu

A list of action options. You select an action by typing its corresponding number in the option input field and pressing Enter. A menu panel might contain other fields that you can use to qualify the action.
migrate

The process of moving DB2 data structures, data structures and data, or data structure changes, from an origin to a destination subsystem.

migrate profile

A profile that can be reused to select a set of objects, to customize changes to objects which are migrated to different locations or used in a comparison, or both.

migrate-type work ID

A work ID that is used to migrate data structures, data, or both. Worklists that are generated from a migrate-type work ID do not contain SQL DROP statements. These work IDs can have migrate options and change rules.

mixed list

A panel in the Specification component that can display multiple DB2 object types.

name template

See “baseline name template” on page 231 and “work ID name template” on page 255.

name propagation

The process of extending to dependent objects the changes that you make in a referenced object. For example, if you change the name of a table that is referenced in a view, name propagation replicates the new name in the view definition.

null

A special value that indicates the absence of information.
object

A DB2 entity that can be created or dropped. Objects are storage groups, databases, table spaces, tables, indexes, foreign keys, views, synonyms, aliases, or triggers. Unlike DB2, which treats foreign keys and check constraints as table attributes, ALTER and CHANGE MANAGER treat foreign keys and check constraints as independent objects.

object list

A list of one or more objects of different types.

option

A named value that is used to control one or more components. Global options are defined in the installation options module. The user can override the installation options by specifying the user options or by specifying keywords in the component’s ALUIN, AEXIN, or AJXIN input stream.

origin

The sending location or source of a migration or a change migration.

orphaned auxiliary index

An auxiliary index that is not associated with an existing auxiliary table.

orphaned auxiliary table space

An auxiliary table space that does not contain an auxiliary table.

outbound migrate profile

A profile that is used in performing a change migration. An outbound migrate profile might define one or more locations that contain change rules that are defined for any or all of those locations. Optionally, an outbound migrate profile can have a scope to select the DB2 objects on which to operate.
P

page

A unit of storage within a table space (4K or 32K) or index space (4K). A page in a table space contains one or more rows of a table. 8K and 16K pages can be used.

parent

A DB2 object that contains the primary key which might be referenced by one or more foreign keys in the child table.

*See also “child” on page 234.*

partitioned data set (PDS)

A data set in direct access storage that is divided into partitions, called members, each of which can contain a program, part of a program, or data. Synonymous with program library.

partitioned table space

A table space that is subdivided into parts (based upon index key range), each of which can be processed independently by utilities.

pattern

A rule that is applied to the naming of objects of a specified type. You can use wildcard characters (% and *) when you define a pattern.

PDS

*See “partitioned data set (PDS)” on page 247.*

privilege

The capability of performing a specific function (authorization) on an object. Privileges might be explicitly or implicitly granted.
profile

A collection of scope rules, change rules, and locations that enables you to define and control a change, migrate, or baseline process.

See also “baseline profile” on page 232, “migrate profile” on page 245, “inbound migrate profile” on page 242, and “outbound migrate profile” on page 246.

protected baseline

A baseline that is designated as protected from deletion. A protected baseline cannot be deleted until the protected designation is removed.

See also “unprotected baseline” on page 253.

R

receive-type work ID

A work ID that is used to create new data structures and load migrated data on a different subsystem.

recovery

The process of restoring a set of data structure definitions to their state at a particular point in time. Recovery involves comparing the DB2 catalog to a baseline, importing the CDL, generating a worklist with Analysis, and executing the worklist. If the baseline is a full-recovery baseline, you can recover data and the data structure definitions.

reference location

A location in a profile whose rules are used by another location. For example, a location called Houston might have rules that are explicitly defined, while locations called Austin and Dallas might use Houston as a reference location.

reference profile

A profile whose scope is used by another profile. A baseline profile can reference a migrate profile, and a migrate profile can reference a baseline profile. Using a reference profile enables you to define the scope only once and thus eliminates errors that might arise from redesigning the scope.
referenced object

An object that a dependent object references. If you change the definition of a referenced object, dependent objects might not continue to function properly.

referential constraint

The requirement that nonnull values of a designated foreign key are valid only if they equal values of the primary key of a designated table. The relationship between the primary key in the parent table and a foreign key in a dependent table is used to establish referential integrity in a database. A referential constraint is always assigned a name to distinguish it from other constraints.

referential integrity

The condition that exists when all intended references from data in one column of a table to data in another column of the same or a different table are valid. Maintaining referential integrity requires enforcing referential constraints on all LOAD, RECOVER, INSERT, UPDATE, and DELETE operations.

S

SBCS

See “single-byte character set (SBCS)” on page 250.

scope

The final set of DB2 objects that are selected from the catalog on which the product operates. A scope includes objects that are explicitly selected and any of their object dependencies. For the Baseline component, the scope selects the objects that are captured in the baseline. For the Compare and Analysis components of CHANGE MANAGER, the scope selects the objects that are included in the comparison or analysis process. A scope consists of one or more scope rules.

scope rule

A specification for selecting DB2 objects from the catalog by object type and name.

Script table
In the CM/PILOT component of CHANGE MANAGER, ordered steps that prompt you for the information that is required to perform a change management task.

**selection list**

A list of related items from which you can select one for further action. The actions (line commands) that you can specify in the Act field are typically displayed across the top of the panel.

**sequence number**

A six-digit, zero-filled number that identifies a statement in a worklist. The sequence number appears in columns 7 through 12 of the first line of each worklist command.

**single-byte character set (SBCS)**

A character set in which each character is represented by a one-byte code.

**source**

In CHANGE MANAGER, the original object of a migration or the original object of a Compare process. When migrating objects or databases, the source database is the database from which you are migrating. The Compare process compares a source data structure with a target data structure. This process synchronizes two data structures, and the source is the data structure that needs to change.

*See also “Compare1” on page 235.*

**Space Estimation**

A feature of ALTER and CHANGE MANAGER that enables you to determine the amount of space that a table space or index will require, based on the object definitions and their estimated usages.

**Specification**

A component of ALTER and CHANGE MANAGER that enables you to create or edit data structure change or migration requests. Specification stores its change or migration requests in a work ID.
SQL

See “Structured Query Language (SQL)” on page 251.

SQLID

The authorization ID that is used as the implicit qualifier of table, view, synonym, and index names in dynamic SQL statements. The SQL ID, along with the other authorization IDs of a process, is used for authorization checking of dynamic SQL statements.

SSID

A DB2 subsystem identifier.

structure-only baseline

A baseline that contains only data structure definitions. No data from those data structures are included.

Structured Query Language (SQL)

An ANSI-standard language for database definition, manipulation, and query.

sub-element

A component of an object. For example, a column is a sub-element of a table, and a volume is a sub-element of a storage group.

symbolic variable

A user interface variable that has its value set interactively for the current user and session at the time of job control language (JCL) generation. The BMC JCL Generation component (JCLGEN) uses symbolic variables to perform ISPF file tailoring services. A symbolic variable should be preceded with an ampersand (&). In the installation options modules, a symbolic variable should be preceded with two ampersands (&&). Symbolic variables should not be confused with global job variables (AJX-type), which have their values set for all users and all sessions.

See also “JCL Generation (JCLGEN)” on page 243.
SYNC

A worklist command that invokes a checkpoint to use for restart processing and commits data to DB2.

sync point

A completion flag that is set during the execution of a worklist. The Execution program writes sync points to the SYNC table whenever it encounters -SYNC or -STOP commands in the worklist input stream. All SQL statements between sync points are executed as a single DB2 transaction. If a worklist is halted before completion for any reason, sync points enable you to begin processing the worklist from the last sync point.

synchronization

The process of identifying structural differences between two copies of the same data structure and then making the data structures identical. For example, separate groups of developers might be independently modifying several copies of a set of data structures. At various times, the copies need to be synchronized to ensure that all of the developers are using the same structure definitions.

T

target

The object of a migration or the object of a Compare process. When migrating objects or databases, the target database is the database to which you are migrating. The Compare process compares a source data structure with a target data structure. This process synchronizes two data structures, and the target is the data structure that contains the wanted changes.

See also “Compare2” on page 235.

task ID

A unit of work in the CM/PILOT component.

template

A method of obtaining specifications for the definition of an auxiliary table space, table, or index. This definition can be replicated for each of the partitions in the base table space.
See also “baseline name template” on page 231 and “work ID name template” on page 255.

U

unit

A specific device, device type, or group of devices that are used in data set allocation.

unload data set

An object that is used to store data while DB2 objects are dropped and rebuilt. The unload data set is also referred to as a SYSREC data set.

unprotected baseline

A baseline that is not designated as protected from deletion.

See also “protected baseline” on page 248.

UOW

Unit of Work. A unit of work consists of the worklist commands that are bounded by the -BEGU and -ENDU commands and that are run in a BMC Cross-System Image Manager (XIM) initiator.

UOWTRnnn

The ddname of the unit of work (UOW) output of the Execution component. When the worklist parallelism feature of the Database Administration solution is used, this output data set contains tracing records for each BMC Cross-System Image Manager (XIM) initiator.

user options

A set of options that are stored in the user’s profile and that are used by Front End and JCL Generation for running ALTER or CHANGE MANAGER components. The user options are initially set from the installation options module the first time that the user runs ALTER or CHANGE MANAGER.
variable

See “symbolic variable” on page 251.

VCAT allocation

A volume placement parameter that specifies the data set high-level qualifier that is appropriate for the DB2 subsystem. Use this field for nonstorage group allocation only.

versioning

The process of comparing baselines that represent the same set of structures at different points in time in order to change a version of a data structure.

wildcard

A symbol that you can use to represent a value in SQL statements, filters, and name patterns. Valid wildcards for SQL statements and filters include the following symbols:

- The % and * represent any character string.
- The _ and ? represent a single character.

work ID

A unit of work with a two-part name (owner.name) that contains change or migration requests in change definition (CD) tables. The change or migration requests can be either imported or created manually through the Specification component.

work ID name

A work ID name is a string of up to eighteen alphanumeric characters excluding percent (%), asterisk (*), underscore (_), and space. When you specify a work ID name, you can use a wildcard pattern to display a group of similar names.
work ID name template

A template from which the name of a work ID is created. You can use this template to create or replace a work ID in batch mode. The name template contains the special character sequences #### and @@@@@@ or @@@@@@@@@, in addition to text characters. An ascending sequence of numbers replaces the #### sequence, and the current date replaces the @@@@@@ or @@@@@@@@@ sequence when the work ID is generated. You can create and replace receive-type work IDs (using name templates) with the Import and Execution components.

work ID owner

The Authorization ID of the creator of the work ID.

worklist

A data set that contains commands for implementing a data structure change or migration.

worklist execution log

See “AEXPRINT” on page 228.

worklist parallelism

A feature in the Database Administration solution that reduces the elapsed time for executing a worklist that is generated by the CHANGE MANAGER product.

X

XIM

See “Cross-System Image Manager (XIM)” on page 236.

XIM initiator

A program that executes one or more units of work (UOW).
Symbols

_ (underscore)
  as a literal 145
  as a wildcard 145

? (question mark) as a wildcard 145

* (asterisk) as a wildcard 145

% (percent sign) as a wildcard 145

$AJXDOC data set 158

A

above-the-bar storage 157
access
  controlling 149
  to database 165
  to parallelism feature, restricting 156

ACT
  definition 227

ACM_PARALLEL_MAXINIT AJXPOFIN keyword 211
ACM_PARALLEL_MININIT AJXPOFIN keyword 211
ACM_PARALLEL_WORKLIST AJXPOFIN keyword 211
ACM_PARALLEL_XIMGRP AJXPOFIN keyword 211
ACM_PARALLEL_XIMPROC AJXPOFIN keyword 211
ACM_PARALLEL_XIMSTRT AJXPOFIN keyword 211
ACM_PARALLEL_XIMTRCE AJXPOFIN keyword 211

ACMvryA Analysis plan 149
ACMvryB Baseline plan 150
ACMvryC Compare plan 150
ACMvryE Environment plan 149
ACMvryF Front End plan 149
ACMvryI Import plan 149

ACMvryR Baseline Report plan 150
ACMvryS Specification plan 149

ACT
  action code
    definition 227

Administrative Assistant for DB2 44
Administrative Products for DB2
  definition 227

AEXAUNLD package 151
AEXEFAUH bit 155
AEXESTDL package 151

AEXIN
  definition 227

AEXIN input stream
  user options 139

AEXPIN
  definition 228

AEXPIN input stream
  using in a parallel worklist 211

AEXPRT
  definition 228

AEXPRnnn
  definition 228

AEXPRTRAC
  definition 228

AEXSQLIO package 151

AEXSYnnn
  definition 228

AEXUSERX
  DSECT 152

AEXvrHA Execution Monitor Entry plan 151
AEXvrHM Execution Monitor plan 151

AJX variables
  definition 228

AJXIN
  definition 228

AJXIN input stream 139

AJXPODAT edit macro 218
AJXPOFER message file 221
AJXPOFIN  
definition 229

AJXPOFVL  
definition 229

AJXPOFVL message file 221

AJXPRINT  
definition 229

aliases, INSERT statement 165

alloc unit  
definition 229

allocating data sets 144

ALTER  
architecture 23
components 25
Execution plans 151
objects 28
overview 15
plans, list of 149

ALTER for DB2  
definition 229

ALTER objects  
DDL file 30
internal table 30
unload data set 30
work ID 28
worklist 29
alter-type work ID 28

definition 229

ALU  
definition 229

ALUDSYNC member 164
ALUEUSX1 member 152

ALUIN  
definition 229

ALUPRINT  
definition 230

Analysis  
definition 230

Analysis component  
default data set name 171
of ALTER 26
of CHANGE MANAGER 33
setting default processing options 171

Application  
definition 230

Application ID 140
applications, user options 179
asterisk (*) as a wildcard 145

ASU  
definition 230

ASXFLAG 155
attribute  
definition 230

AUTHID  
definition 230

authorization  
setting 137

authorization ID  
definition 230

authorization switching  
controlling 152

auxiliary index  
definition 230

auxiliary list  
definition 230

auxiliary table  
definition 231

auxiliary table space  
definition 231

baseline component  
default data set name 177
setting default processing options 177

Baseline component  
of CHANGE MANAGER 33

baseline name template  
definition 231

baseline profile  
definition 232

Baseline Report component  
default data set name 178
setting default processing options 178

baselines  
default run type 177
object 37
profile 38

Table maintenance 164
table sizing and growth 164
worklist 37

batch component
<table>
<thead>
<tr>
<th>Definition</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>BL_ALIAS product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_AUXTABLE product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_CHECK product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_CHECK2 product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_COLUMN product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_DATABASE product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_DBAUTH product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_KEY product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_KEYTARGETS product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_PACKAGE product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_PARMS product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_REL product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_RELKEY product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_RESAUTH product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_ROUTINES product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_ROUTINESTEXT product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_RTAUTH product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_SEQUENCES product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_SQAUTH product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_STOGROUP product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_SYNONYM product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_TABLE product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_TABLEPART product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_TABLESPACE product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_TRIG_COL product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_TRIG_TEXT product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_TRIG_TEXT_AUX product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_TRIGGER product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_UNQCNST product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_UNQCNST_COL product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_VIEW product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_VIEW_TEXT product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_VIEW_TEXT_AUX product table</td>
<td>160</td>
</tr>
<tr>
<td>BL_VOLS product table</td>
<td>160</td>
</tr>
<tr>
<td>BLRP data set</td>
<td>195</td>
</tr>
<tr>
<td>BMC object</td>
<td>232</td>
</tr>
<tr>
<td>BMC Object Administration for DB2</td>
<td>45</td>
</tr>
<tr>
<td>BMC Software solutions</td>
<td></td>
</tr>
<tr>
<td>Administrative Assistant for DB2</td>
<td>44</td>
</tr>
<tr>
<td>BMC Object Administration for DB2</td>
<td>45</td>
</tr>
<tr>
<td>Database Administration for DB2</td>
<td>44</td>
</tr>
<tr>
<td>BMCADMF2 CLIST</td>
<td>50</td>
</tr>
<tr>
<td>BMCALTER command</td>
<td>50</td>
</tr>
<tr>
<td>BMCCAT command</td>
<td>50</td>
</tr>
<tr>
<td>BMCCCHG command</td>
<td>50</td>
</tr>
<tr>
<td>BMCCOPY</td>
<td>232</td>
</tr>
<tr>
<td>BMCDASD command</td>
<td>50</td>
</tr>
<tr>
<td>BMCD2 CLIST</td>
<td>140</td>
</tr>
<tr>
<td>BMCLOAD</td>
<td>232</td>
</tr>
<tr>
<td>BMCREORG</td>
<td>232</td>
</tr>
<tr>
<td>BMCSSTATS</td>
<td>232</td>
</tr>
<tr>
<td>BMCUNLOAD</td>
<td>233</td>
</tr>
<tr>
<td>CAN</td>
<td>233</td>
</tr>
<tr>
<td>CANCEL</td>
<td>233</td>
</tr>
<tr>
<td>CANCEL ISPF command</td>
<td>51</td>
</tr>
<tr>
<td>carriage return, hexadecimal format</td>
<td>148</td>
</tr>
<tr>
<td>CASE tools</td>
<td>22</td>
</tr>
<tr>
<td>catalog baseline</td>
<td>233</td>
</tr>
<tr>
<td>catalog baseline profiles, user option for CM/PILOT</td>
<td>179</td>
</tr>
<tr>
<td>catalog indirection</td>
<td>233</td>
</tr>
<tr>
<td>CATALOG MANAGER for DB2</td>
<td>233</td>
</tr>
<tr>
<td>CD_ALIAS product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_AUXTABLE product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_CHECK product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_CHECK2 product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_COLUMN product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_DATABASE product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_INDEX product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_KEY product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_KEYTARGETS product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_PACKAGE product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_PARMS product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_REL product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_RELKEY product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_ROUTINES product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_ROUTINES2 product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_ROUTINES2NEWTEXT product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_ROUTINES2TEXT product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_SEQUENCES product table</td>
<td>160</td>
</tr>
<tr>
<td>CD_STOGROUP product table</td>
<td>160</td>
</tr>
</tbody>
</table>
CD_SYNONYM product table 160
CD_TABLE product table 160
CD_TABLEPART product table 160
CD_TABLESPACE product table 165
CD_TRIG.COL product table 160
CD_TRIG_TEXT product table 160
CD_TRIG_TEXT_AUX product table 160
CD_TRIGGER product table 160
CD_UNQCNST product table 160
CD_UNQCNST_COL product table 160
CD_VIEW product table 160
CD_VIEW_TEXT product table 160
CD_VIEW_TEXT_AUX product table 160
CD_VOLS product table 160

CDL
definition 233

CDL (Change Definition Language)
data set name, specifying 176
data set, allocating 144

Change Definition Language CDL
definition 233

CHANGE MANAGER
architecture 30
components 33
database access 165
Execution plans 151
objects 36
overview 36
plans, list of 149

CHANGE MANAGER for DB2
definition 234

CHANGE MANAGER objects
application 43
baseline 37
CDL file 42
CM/PILOT worklist 43
DDL file 42
DML 43
internal table 42
profile 38
script 43
task ID 43
unload data set 30
work ID 28
worklist 29
change migration process 20
change rule
definition 234
change rules, used in outbound migrate profiles 41
changes
managing with ALTER 15
characters
delimited identifiers 147
nonprintable 148
check constraints, INSERT statement 165
checkpoint
definition 234
child
definition 234
cleanup, data set 185

CLIST
BMCADMF2 50
BMCDB2 140
definition 234

CM_APPLICATION product table 160
CM_BASELINE product table 160
CM_BLPROFILE product table 160
CM_HISTORY product table 160
CM_LOCATION product table 160
CM_MIGPROFILE product table 160
CM_RENAME product table 160
CM_RULES product table 160
CM_SCOPE product table 160
CM_SECURITY product table 160
CM_SYNC product table 160
CM_TASKID product table 160
CM_UNLOADDSN product table 160
CM_WORKID product table 160
CM_XML product table 160
CM_XML_AUX product table 160
CM/PILOT component
of CHANGE MANAGER 35
CM/PILOT user options
catalog baseline profile 179
CDL 179
changing 179
data sets 179
DDL baseline profile 179
processing a worklist 179
worklist 179
CM/PILOT worklist 43
CMPILOT for DB2
definition 234
CMPILOT worklist
definition 234
CMvryL LOB Data Mover plan 150
columns, INSERT statement 165
command
definition 235
commands
D

DASD MANAGER PLUS
  Execution plans 151
DASD MANAGER PLUS for DB2
  definition 236
Data Control Language DCL
  definition 236
data conversions, using in change process 18
data definition language DDL
  definition 236
data definition name ddname
  definition 236
Data Manipulation Language DML
  definition 236
data modeling tools, using 22
data set
  map 185
data set sizing
  definition 236
data sets

DASD MANAGER PLUS
  Execution plans 151
DASD MANAGER PLUS for DB2
  definition 236
Data Control Language DCL
  definition 236
data conversions, using in change process 18
data definition language DDL
  definition 236
data definition name ddname
  definition 236
Data Manipulation Language DML
  definition 236
data modeling tools, using 22
data set
  map 185
data set sizing
  definition 236
data sets

commit
definition 235
Compare
definition 235
Compare component
default data set name 176
of CHANGE MANAGER 34
Compare1
definition 235
Compare2
definition 235
comparisons
  primary input 34
  secondary input 34
  setting default processing options 176
component
definition 235
components
  Analysis 26
  Baseline 33
  CM/PILOT 35
  Compare 34
  Execution 27
  Front End 25
  Import 26, 35
  JCL Generation 28
  Specification 25
constraint
definition 235
control characters, using in object names 148
control point 33
conventions, documentation 10
CP_DML_STMT product table 160
CP_SCRIPT product table 160
CP_STEP product table 160
creating
  additional installation options module 142
  objects 54
Cross-System Image Manager. See XIM
Cross-System Image Manager XIM
  definition 236
$AJXDOC 158
CDL 144
cleanup 185
DATAWORK 193
deleting automatically 185
diagnostic output 144
discard 185
error 185
JCL 144
LOGSORT 193
permanent work 195
prefixes 195
punch 185
report 144
resolving names 195
sample VSAM object 185
setting default processing options 195
setting installation options 170
sizing 159, 185
SORTOnnn 185
SORTOUT 185, 195
SORTPnnn 185
SORTWORK 185, 193
SUTnnn 185
SYSCOPY 195
SYSDISC 195
SYSERR 195
SYSMAP 195
SYSPUNCH 195
SYSREC 195
SYSUT 195
SYSUTnnn 185
unload 185
user options, task ID 179
work 185
worklist 144
WRKnnn 185
data structure
definition 237
data structures, importing 18
Database Administration for DB2 44
database administrator DBA
definition 237
database request module DBRM
definition 237
databases, INSERT statement 165
DATASETSIZING B keyword 187
DATASETSIZING C keyword 187
DATASETSIZING O keyword 187
DATAWORK data set 193
DB2 catalog
definition 237
DB2 command
definition 237
DB2 objects, security 165
DB2 SSID, specifying 170
DB2 utilities
used by ALTER 15
used by CHANGE MANAGER 15
DBA
definition 237
DBCS
definition 237
DD statement
definition 237
DDL
definition 237
DDL baseline
definition 238
ddname
definition 238
defaulting, setting default processing options 205
default options. See installation options
default options module
definition 238
default output class 214
default value
definition 238
defaults, setting product options
data sets 195
debugging 205
generation data groups 203
JCL jobcard 182
JCL static data sets 185
JCL STEPLIBs 184
product options file 217
SORTWORK data sets 193	
utility module names 207
defaults, setting user options
Analysis options 214
Analysis processing 171
baseline processing 177
Baseline Report processing 178
Compare processing 176
data set allocation 170
data sets 195
debugging 203
Execution processing 174
general 137
generation data groups 203
Import processing 173
online reorgs 208
screen display 168
SHRLEVEL CHANGE 208
user variables 216
worklist parallelism 211
worklist sorting 213
deleting
baselines 164
data sets automatically 185
task IDs 164
work IDs 164
delimited identifier
definition 238
delimited identifiers
valid and invalid wildcard use 145
wildcard support 147
dependencies
definition 238
dependent
definition 238
dependent object
definition 238
destination
definition 238
device type
definition 239
diagnostic output
data sets 144
default data set name for Baseline 177
default data set name, for Analysis 171
default data set name, for Baseline Report 178
default data set name, for Compare 176
default data set name, for Execution 174
default data set name, for Import 173
discard data set 185
documentation information 9
DOPTS
definition 239
DOPTS command 138, 142
double-byte character set DBCS
definition 239
DOWN ISPF command 51
DSECTs
AEXUSERX 152
duplicate
definition 239
dynamic allocation
data set prefixes 195
diagnostic output data set 144
setting options 213
edit procedure
definition 239
EDITPROC
definition 239
electronic documentation 9
END
definition 239
END ISPF command 51
enquiry character, hexadecimal format 148
ENTER ISPF command 52
Enter key
definition 240
ENVI command 49
ENVIRONMENT command 49
error data set 185
Execution
definition 240
Execution component
default data set name 174
of ALTER and CHANGE MANAGER 27
of DASD MANAGER PLUS 27
packages 151
plans, list of 151
security exit testing 155
security exit, used by 152
Execution plans 149
execution, worklist
authorization switching 152
cleanup job step for data sets 185
default run type 176
deleting data sets automatically 185
overriding defaults for worklist parallelism 211
setting default processing options 174
exit routine
definition 240
external hexadecimal format 148
external stored procedure, INSERT statement 165

F
F1 (HELP) 52
F10 (LEFT) 52
F11 (RIGHT) 52
F12 (PREV) 52
F2 (SPLIT) 52
F3 (END) 51
F4 (ZOOM) 53
F7 (UP) 52
F8 (DOWN) 51
F9 (SWAP) 52
FCPY_SUPPRESS_SUFF POF keyword 195
field procedure
definition 240
FIELDPROC
definition 240
file tailoring 158
fix
using the MAINT command 49
foreground component
definition 240
foreign key, data set sizing 159
foreign keys, INSERT statement 165
form feed character, hexadecimal format 148
Front End
definition 240
Front End component
description 25
generating input streams 141
full-recovery baseline
definition 241

g
GDG
definition 241
GDG (generation data group)
setting default processing options 203
specifying 195
symbolic variable 195
using for data sets 195
GDG symbolic variable 195
general resource profile 157
generation data group. See GDG
generation data group GDG
definition 241
GLID (global authorization ID), default 171
Group ID
definition 241

H
hash value
definition 241
Help
online 9
HELP ISPF command 52
HLQ
definition 241

I
IEFUSI exit 157
image copy
definition 242
import
definition 242
Import
definition 242
Import component
default data set name 173
of CHANGE MANAGER 26, 35
importing
default run type 173
setting default processing options 173
inbound migrate profile
definition 242
incremental DDL
definition 242
indexes, INSERT statement 165
initial POF, refreshing 220
input streams 139
installation options
creating 142
user options 141
installation options module
definition 242
internal table
definition 242
ISPF
AJXPODAT macro 218
commands 51
file tailoring, used by JCL Generation 158
help 52
ISPF interface
  Main Menu 48
  using 47
ISPF profile
  user options 141
ISPF skeletons
  definition 243

J
JCL
data set, allocating 144
  definition 243
JCL data set name
  Analysis 171
  Baseline Report 178
  Compare 176
  Execution 174
  Import 173
JCL DSN
  definition 243
JCL Generation
data set sizing 159
  product options 140
  RUNSTATS utility 159
  VSAM object sampling 159
JCL Generation component 28
JCL Generation JCLGEN
  definition 243
JCL job card, setting default processing options 182
JCL job statement, defaults 182
JCL static data sets, setting default processing options 185
JCL STEPLIBs, setting default processing options 184
JCL variable display
  definition 243
JCLGEN
  definition 243
job
  definition 243

L
large object LOB column
definition 244
large object LOB table space
  definition 244
LEFT ISPF command 52
LINK library
  definition 244
literal strings 148
LOB DATA MOVER
  definition 244
LOB SYSREC data set 195
  location
    definition 244
  locations, using 41
  locking options 50
LOGSORT data set 193

M
macros
  AJXPODAT 218
MAINT command 49
  maintaining, sync tables 164
  maintenance
    MAINT command 49
    managing change
      using ALTER 15
      using CHANGE MANAGER 19
map data set 185
mapping database 208
mapping table 208
MAXINITS AEXPIN keyword 211
MEMLIMIT system parameter 157
menu
  definition 244
message files 221
message identifier 53
migrate
  definition 245
migrate profile
  definition 245
migrate-type work ID
  definition 245
migration
  inbound migrate profile 179
  objects 86
MININITS AEXPIN keyword 211
mixed list
  definition 245
  modifying, user options 179
msg_ID 53
multitasking, using tape stacking 191

N
name propagation
definition 245
name template
definition 245
naming objects 144
native SQL stored procedure, INSERT statement 165
negative acknowledgegment, hexadecimal format 148
NEXT ISPF command 52
nonprintable characters 148
nonviewable characters 148
null
definition 245
null character, hexadecimal format 148
null characters 148

O
object
definition 246
object list
definition 246
objects
ALTER 28
creating 54
migrating 86
naming 144
security 144
online Help 9
online reorgs 208
option
definition 246
options
Analysis 171
baseline 177
Baseline Report 178
changing 137
CM/PILOT 179
Compare 176
data set allocation 170
data sets 195
descking and display 205
Execution 174
generation data group 203
Import 173
installation 138
JCL job cards 182
JCL static data sets 185
JCL STEPLIBs 184
online reorg 208
panel display 168
product options file 217
refreshing 139
setting 137, 168
SHRLEVEL CHANGE 208
SORTWORK data sets 193
tapes 191
used by products 143
user 139
user variables 216
utility module names 207
worklist parallelism 211
origin
definition 246
orphaned auxiliary index
definition 246
orphaned auxiliary table space
definition 246
outbound migrate profile
definition 246
outbound migrate profiles, used by CHANGE MANAGER 39
override options, using keywords 141

P
packages, Execution component 151
page
definition 247
panel display 168
PARALLEL AEXPIN keyword 211
parameters
MEMLIMIT 157
system 157
parent
definition 247
partitioned data set PDS
definition 247
partitioned table space
definition 247
pattern
definition 247
PCPY1_SUPPRESS_SUFF POF keyword 195
PCPY2_SUPPRESS_SUFF POF keyword 195
PDS
definition 247
percent character, hexadecimal format 148
percent sign (%) as a wildcard 145
permanent work data sets 195
PFSHOW ISPF command 52
plans
ALTER 149
authorizations 149
CHANGE MANAGER 149, 150
Execution component 151
names 149
security 149
synonyms 49
POF
AJXPODAT macro 218
creating a user POF 217
setting default processing options 217
updating a user POF 218
POF (product options file)
generating reports 221
initial 140
initializing 140
overriding values in SLIBs 223
POF Validation Report 221
populating 140
refreshing the initial POF 220
reusing in a subsequent installation 222
user 140
using multiple POFs 219
Variables Initialized with Default report 221
POFDATE POF keyword 140
POFDS installation option 140
prefixes 195
PREV ISPF command 52
privilege
definition 247
product database 165
product options file. See POF
product tables
list of 160
maintaining 160
sizing 164
profile
definition 248
profiles
baseline 38
catalog baseline 38
DDL baseline 38
inbound migrate 39
migrate 39
outbound migrate 39
user options, application 179
propagating, changes 14
protected baseline
definition 248
publications, related 9
punch data set 185

Q
question mark (?) as a wildcard 145

R
RACF (IBM Resource Access Control Facility) 157
RCPY1_SUPPRESS_SUFF POF keyword 195
RCPY2_SUPPRESS_SUFF POF keyword 195
receive-type work ID
definition 248
recovery
definition 248
reference location
definition 248
reference profile
definition 248
referenced object
definition 249
referential constraint
definition 249
referential integrity
definition 249
refreshing user options 139
related publications 9
rename table
security 165
updating 165
report data set 144
reports
  setting installation options 178
reports, generating POF 221
restricting access
  specifying a resource profile 157
to worklist parallelism 156
RIGHT ISPF command 52
ROWID SYSREC data set 195
RUNSTATS utility 159

S

SBCS
  definition 249
scope
  definition 249
scope rule
  definition 249
scope rules, CHANGE MANAGER object 40
scope, used by Compare 40
Script table
  definition 249
script, CM/PILOT 35
security
  BMC object 144
  component 149
  DB2 objects 165
  plan 149
  rename table 165
security authorization 152
security exit
  Execution exit 152
  testing the execution 155
  used by Execution 152
security, setting authorizations 137
selection list
  definition 250
sequence number
  definition 250
sequences, INSERT statement 165
SHRLEVEL CHANGE 208
Simple Space Estimation (SSE) feature 52
single-byte character set SBCS
  definition 250
skeleton library, overriding POF values 223
SMFPRM member 157
SMS (Storage Management Subsystem) 147, 195
SORTOnnn data set 185

SORTOUT data set 185, 195
SORTPnnn data set 185
SORTWORK data set 185
source
  definition 250
Space Estimation
  ALTER 19
  CHANGE MANAGER 25
  definition 250
Specification
  definition 250
Specification, component of ALTER and CHANGE
  MANAGER 25
SPLIT ISPF command 52
SQL
  definition 251
SQLID
  definition 251
SSE ISPF command 52
SSID
  definition 251
STEPLIB libraries, setting default values 182
storage group, INSERT statement 165
Storage Management Subsystem. See SMS
storage, virtual 157
structure-only baseline
  definition 251
Structured Query Language SQL
  definition 251
sub-element
  definition 251
SUTnnndata set 185
SWAP ISPF command 52
Switching authorization 152
symbolic variable
  definition 251
symbolic variables
  GDG 195
SYNC
  definition 252
sync point
  definition 252
sync table 164
synchronization
  definition 252
synonyms
  ENVI command 49
  INSERT statement 165
syntax statement conventions 10
SYSCOPY data set, setting JCL options 195
SYSDISC data set 195
SYSERR data set 195
SYSMAP data set 195
SYSOUT 214
SYSPUNCH data set 195
SYSREC data sets
  LOB 195
  ROWID 195
  setting JCL options 195
SYSUT data set 195
SYSUTnnn data set 185

table spaces, INSERT statement 165
tables
  baseline 164
  column, INSERT statement 165
  INSERT statement 165
  maintenance 160
  security 165
  sizing 164
tapes
  setting default processing options 191
  stacking, disabling 191
  using 195
target
  definition 252
task ID
  definition 252
task IDs
  CHANGE MANAGER object 43
  deleting 164
  user options 179
Task List 54, 86
template
  definition 252
temporary work data sets 193
threshold, for alternate unit 195
TRACE AEXPIN keyword 211
triggers, INSERT statement 165
TSO region 47
TSO region size 137

underscore (_)
  as a literal 145
  as a wildcard 145
unique constraints, INSERT statement 165
  unit
    definition 253
  unit name, ROWID SYSREC data set 195
UNLD1_SUPPRESS_SUFF POF keyword 195
UNLD2_SUPPRESS_SUFF POF keyword 195
unload data set
  definition 253
unload data set, used by utilities 185
unprotected baseline
  definition 253
UOW
  definition 253
UOWTRnnn
  definition 253
UP ISPF command 52
  definition 253
user exits
  Execution security 152
user options
  CDL 179
  changing CM/PILOT 179
  definition 253
  general 137
  JCL Generation 140
  override options 141
user POF
  creating 217
  updating directly 218
  updating in options panels 218
  using multiple 219
user variables 216
utility module names, setting default processing options 207

values
  hexadecimal 177
  nonprintable characters, using 148
variable
  definition 254
VCAT allocation
  definition 254
versioning
  definition 254
view column, INSERT statement 165
viewing, INSERT statement 165
virtual storage 157
VSAM object
| A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
| data set sizing | 185 | JCL generation | 159 | sampling | 159 |
| W | wildcard | definition | 254 |
| wildcard characters | delimited identifier support | 147 |
| used with change rules | 145 |
| used with scope rules | 145 |
| used with Specification filters | 145 |
| valid and invalid wildcards | 145 |
| work data sets | permanent | 195 |
| temporary | 193 |
| used by utilities | 185 |
| work ID | definition | 254 |
| work ID name | definition | 254 |
| work ID name template | definition | 255 |
| work ID owner | definition | 255 |
| work IDs | deleting | 164 |
| name template | 179 |
| worklist | definition | 255 |
| worklist execution log | definition | 255 |
| worklist parallelism | definition | 255 |
| overriding defaults | 211 |
| setting default processing options | 211 |
| worklists | data set | 144 |
| DDL baseline | 37 |
| overview | 29 |
| setting default data set name | 171 |
| used by Baseline | 37 |
| WRKnnn data set | 185 |
| X | XIM | definition | 255 |
| XIM (Cross System Image Manager) | 156 |
| XIMGROUP AEXPIN keyword | 211 |
| XIMPRC AEXPIN keyword | 211 |
| XIMSTART AEXPIN keyword | 211 |
| Z | zaps | applying to restrict access | 156 |
| using the MAINT command | 49 |
| ZOOM command | 168 |
| ZOOM ISPF command | 53 |